batteries

IMPACT FACTOR 4.8

CITESCORE 6.6

Article

Experience-Driven NeuroSymbolic System for Efficient Robotic Bolt Disassembly

Pengxu Chang, Zhigang Wang, Yanlong Peng, Ziwen He and Ming Chen

Special Issue

Batteries: 10th Anniversary

Edited by

Prof. Dr. Karim Zaghib

MDPI

https://doi.org/10.3390/batteries11090332 batteries

MDPI

Article

Experience-Driven NeuroSymbolic System for Efficient Robotic Bolt Disassembly

Pengxu Chang [1], Zhigang Wang [2], Yanlong Peng [1], Ziwen He [1] and Ming Chen [1,*]

[1] School of Mechanical Engineering, Shanghai Jiaotong University, No. 800, Dongchuan Road, Shanghai 200240, China; 19119175490@sjtu.edu.cn (P.C.); me-pengyanlong@sjtu.edu.cn (Y.P.); heziwen@sjtu.edu.cn (Z.H.)
[2] Vision and AI Lab, Intel Labs China, the 8F, Raycom InfoTech Park Tower A, No. 2, Kexueyuan Nanlu, Beijing 100190, China; zhi.gang.wang@intel.com
* Correspondence: mingchen@sjtu.edu.cn

Abstract

With the rapid growth of electric vehicles, the efficient and safe recycling of high-energy battery packs, particularly the removal of structural bolts, has become a critical challenge. This study presents a NeuroSymbolic robotic system for battery disassembly, driven by autonomous learning capabilities. The system integrates deep perception modules, symbolic reasoning, and action primitives to achieve interpretable and efficient disassembly. To improve adaptability, we introduce an offline learning framework driven by a large language model (LLM), which analyzes historical disassembly trajectories and generates optimized action sequences via prompt-based reasoning. This enables the synthesis of new action primitives tailored to familiar scenarios. The system is validated on a real-world UR10e robotic platform across various battery configurations. Experimental results show a 17 s reduction in average disassembly time per bolt and a 154.4% improvement in overall efficiency compared with traditional approaches. These findings demonstrate that combining neural perception, symbolic reasoning, and LLM-guided learning significantly enhances robotic disassembly performance and offers strong potential for generalization in future battery recycling applications.

Keywords: neurosymbolic; autonomous robotic disassembly; battery recycling; large language model (LLM); rapid and efficient battery pack disassembly

check for updates

Academic Editor: Sylvain Franger

Received: 14 July 2025
Revised: 20 August 2025
Accepted: 2 September 2025
Published: 5 September 2025

Citation: Chang, P.; Wang, Z.; Peng, Y.; He, Z.; Chen, M. Experience-Driven NeuroSymbolic System for Efficient Robotic Bolt Disassembly. *Batteries* **2025**, *11*, 332. https://doi.org/10.3390/batteries11090332

1. Introduction

With the rapid growth in electric vehicle (EV) adoption, a substantial number of power batteries are expected to reach the end of their service life within the next 5 to 10 years. According to the International Energy Agency (IEA), the total volume of retired batteries is projected to exceed several million tons globally by 2030. These used batteries contain large quantities of valuable metals and critical minerals, such as lithium, cobalt, and nickel, as well as integrated sensors, circuits, and other electronic components [1]. Efficient recovery of these resources not only alleviates upstream material constraints and reduces dependence on primary raw materials but also enhances the resilience and sustainability of the EV supply chain.

As the number of decommissioned batteries increases, the recycling of lithium-ion power batteries faces growing technical challenges. Significant variability in battery chemistries (e.g., LFP, LCO, NCA, LMO, and NMC) and cell formats (cylindrical, prismatic, and pouch cells), combined with complex battery pack structures, comprising modules, support frames, high-voltage harnesses, battery management systems (BMSs), thermal management systems, and other electronic units, further complicates the recycling process [2]. Disassembly serves as the first step in physical separation, breaking down battery packs into modules or cells to enable subsequent chemical recovery. Given the structural complexity of different battery packs, multiple specialized disassembly tools are typically required. As a result, efficient disassembly is critical to both second-life applications and material recovery of end-of-life EV batteries [3].

Currently, disassembly of EV power batteries is predominantly manual, to accommodate the unpredictable geometries of retired battery packs. However, in regions with high labor costs, manual disassembly is economically unsustainable, while in lower-cost regions, technicians often lack standardized training and procedures. As the volume of retired batteries increases, manual methods face serious limitations in both efficiency and safety, rendering them unsuitable for large-scale recycling. In response, researchers have begun exploring semi-automated disassembly solutions, such as robotic platforms for repetitive tasks like bolt removal and module separation, to improve throughput and reduce human risk [4].

While human–robot collaboration offers advantages in handling the complexity and uncertainty of battery pack disassembly, achieving fully autonomous and intelligent disassembly remains a major open challenge [5]. To address this, the NeuroSymbolic AI community has proposed a deeply integrated framework for battery disassembly [6–9]. This framework combines continuous perception through neural networks with discrete symbolic reasoning using two core components, neural predicates and action primitives, enabling autonomous scene understanding and decision making.

In practice, the system first captures high-resolution RGB-D images of the battery pack surface using an Intel RealSense D455 depth camera (Intel Corporation, Cupertino, CA, USA). A YOLOv9-based model [10] within the neural predicate module detects bolt targets, which are then refined via a Kalman filter to achieve sub-millimeter 3D pose estimation. Once the preconditions defined in a Planning Domain Definition Language (PDDL) domain are met, a scheduler sequentially activates the corresponding action primitives to execute automated bolt removal. This workflow is detailed in Section 3.

Despite the increasing autonomy achieved by current NeuroSymbolic frameworks, a critical limitation remains: the symbolic space is static and lacks the ability to adapt through continuous learning. Neural predicates and action primitives are typically manually defined via PDDL and remain fixed during runtime, limiting the system's ability to generalize to new environments or improve performance based on experience. This rigidity significantly hinders long-term adaptability and efficiency in managing structurally diverse and dynamically changing disassembly tasks.

To overcome this limitation, we argue that an ideal disassembly robot should exhibit human-like learning capabilities: it should initially rely on logical planning to complete tasks and subsequently refine its strategies and action representations through accumulated experience, thereby improving accuracy, efficiency, and robustness over time.

In response, this paper proposes a novel NeuroSymbolic embodied-intelligence framework augmented with a self-learning mechanism that enables the system to extract and adapt from historical disassembly experiences. Built upon conventional perception, reasoning, and execution modules, our system introduces an LLM-driven adaptive optimization mechanism that enables real-time adjustment and continuous evolution of both neural predicates and action primitives. To validate our approach, we select screw disassembly in electric vehicle battery (EVB) packs as a representative high-frequency task and conduct extensive real-world experiments.

The main contributions of this work are summarized as follows:

- We propose a NeuroSymbolic framework with offline learning capabilities, integrating neural predicates, action primitives, and LLM-based policy adaptation for unified perception, reasoning, execution, and learning.
- We design an LLM-driven adaptive optimization module that dynamically refines execution strategies and decision-making logic, improving the generalization and reusability of action primitives and neural predicates.
- We develop and deploy the proposed system in real-world EVB disassembly scenarios, demonstrating significant improvements in fastener localization accuracy, disassembly efficiency, and task success rate over conventional methods.
- We show that the proposed framework generalizes to other structured industrial disassembly tasks, such as fan units and power modules, establishing a theoretical and practical foundation for embodied intelligence in complex physical environments.
- We enable historical experience-driven learning by allowing the system to autonomously extract knowledge from past disassembly trajectories and iteratively refine disassembly actions, addressing the limitations of traditional systems that lack adaptive memory and self-improvement capabilities.

The remainder of this paper is organized as follows: Section 2 reviews related work in intelligent and automated EVB disassembly. Section 3 presents the system architecture and key components of the proposed framework. Section 4 details the self-learning mechanism for screw disassembly. Section 5 reports experimental results and system validation in realistic EVB disassembly settings. Section 6 concludes the paper and outlines future research directions.

2. Literature Review

To enable safe, efficient, and intelligent disassembly of electric vehicle battery (EVB) packs, recent research has explored automated disassembly systems, autonomous robotic manipulation, and learning-based task planning. For clarity, this review is organized into three closely related themes:

- Section 2.1 surveys the state of the art in EVB disassembly methods, ranging from mechanized and human–robot collaborative systems to emerging intelligent approaches, and discusses their advantages and limitations in real-world applications.
- Section 2.2 reviews recent efforts to apply LLMs to robotic task understanding, planning, and action generation, highlighting their growing role in semantic reasoning and adaptive decision making.
- Section 2.3 focuses on *neural predicate and action primitive learning*, covering NeuroSymbolic representations, imitation and reinforcement learning, and other core techniques that support motion generalization and strategic optimization in dynamic environments.

By synthesizing progress across these three areas, this review identifies current achievements and open challenges and provides the theoretical foundation and research motivation for the self-learning EVB disassembly framework proposed in this paper.

2.1. Progress in Electric Vehicle Battery (EVB) Disassembly Methods

In EVB recycling, disassembly is a crucial and indispensable initial step [11], particularly for mechanical recovery methods that aim to enhance safety while improving the purity of recovered materials. Due to the diversity of EVB types and the uncertainty of their end-of-life conditions, mechanical disassembly still heavily depends on manual labor [12]. This reliance exposes workers to potential hazards, including high voltage, toxic chemicals, fire, and electric arcs [13].

To address these risks, researchers have explored safer, more efficient, and flexible automated disassembly solutions. For example, task-oriented programming via intuitive user interfaces has been proposed as an alternative to conventional teach-in and offline programming approaches, aiming to reduce programming complexity and enable non-expert users to control robots more effectively [14].

Human–robot collaborative systems have also been developed [15–17], where human operators handle highly variable connectors and flexible cable harnesses, while robots focus on repetitive or hazardous tasks. Although these systems address several technical barriers, safety concerns persist.

Duan et al. [18] integrated depth cameras, RANSAC plane fitting, and Kalman filters to accurately detect screw positions and poses in robotic disassembly systems, enhancing flexibility in human–robot collaboration. Additionally, artificial intelligence (AI) and machine learning (ML) techniques have been introduced at various stages of the disassembly pipeline [19,20], including component sorting, safety monitoring, decision making, and target detection, improving the system's capability to manage complex tasks.

Despite these advancements, many current systems are tailored to specific battery types, lack generalization capabilities, and depend heavily on hand-crafted rules and rigid procedures. Consequently, they struggle to adapt to novel or uncertain structural conditions and are unable to learn effectively from prior execution experiences.

2.2. Applications of LLMs in Robotics

In recent years, LLMs, such as the GPT series [21], Palm [22], and Llama [23], have achieved remarkable performance in natural language understanding, semantic reasoning, and text generation. With growing abilities in reasoning and generalization, LLMs have become a promising component in robotics, serving as a bridge between high-level task comprehension and low-level action execution [24,25].

One of the key advantages of LLMs lies in their ability to encode general-purpose knowledge and perform complex semantic reasoning. Researchers have begun using natural language as a task interface, enabling robots to understand and execute human instructions. For example, Google's SayCan framework [26] utilizes a language model to translate user commands into high-level action plans, which are then filtered by a value function to ensure feasibility, thereby linking natural language and robotic control. Similarly, the PaLM-E architecture [27] integrates LLMs into a unified multimodal module for perception, reasoning, and control, enabling end-to-end robotic task execution.

In industrial applications such as disassembly, assembly, and material handling, LLMs have been adopted as auxiliary reasoning engines. These models analyze historical data and generate task optimization suggestions. For instance, Singh et al. [28,29] introduced structured prompting techniques that encode executable robot actions and example scripts, allowing LLMs to generate context-aware and valid task plans. These studies demonstrate the potential of LLMs to assist in generating complex action sequences, diagnosing abnormal states, and replanning in dynamic environments.

Furthermore, LLMs exhibit strong capabilities in generating action primitives, abstract task representations, and inferring user intent. Several works have investigated automatic generation of PDDL models or translation of task descriptions into executable sequences [30,31], thereby lowering the barrier to robotic deployment and enhancing generalization and interpretability.

Although current applications of LLMs in robotics are predominantly offline and focused on planning or advisory functions, their inference capabilities have become increasingly reliable due to advancements in structured prompting and optimization. As techniques like few-shot learning and structured outputs mature, LLMs are expected to play a central role in autonomous task understanding, planning, and knowledge transfer [32]. Their integration with NeuroSymbolic systems is anticipated to significantly boost robotic adaptability and generalization in complex real-world scenarios.

2.3. Autonomous Learning of Neural Predicates and Action Primitives

Neural predicates and action primitives are core components of NeuroSymbolic systems for EVB disassembly. In dynamic and uncertain environments, fixed action scripts are insufficient to address the diversity of real-world scenarios. Therefore, it is essential to develop systems with adaptive and generalizable learning capabilities to achieve high-level robotic intelligence.

Traditional task planning frameworks, such as Stanford Research Institute Problem Solver (STRIPS) [33] and PDDL, rely on manually designed predicate sets. Although these methods offer reliability, they require extensive domain knowledge and labor-intensive annotations, limiting their scalability. To address these issues, recent studies have proposed learning-based predicate induction methods. For instance, Silver [34] introduced a framework that learns task-relevant predicates from demonstration data, supporting automatic predicate generation and improving planning efficiency. However, in complex environments, these methods often require large-scale demonstrations to ensure accuracy, posing challenges for practical deployment.

Other approaches seek to derive structured representations from unstructured inputs. For example, Martin and Doumas [35,36] employed Long-Term Memory (LTM) and Discovery of Relations by Analogy (DORA) architectures to abstract functional predicates from raw demonstrations. While effective in controlled environments, their generalization to real-world tasks remains limited. More recently, Han et al. [37] proposed an LLM-driven interactive framework where predicates are expressed as Python functions and iteratively refined through feedback, enabling adaptability to open-ended and evolving task contexts.

In parallel, action learning has garnered increasing interest. Imitation learning allows robots to learn trajectories or policies by observing expert demonstrations [38,39]. OpenAI's one-shot imitation framework [40] enables generalization from a single demonstration using behavior encoders and policy decoders. Reinforcement learning, on the other hand, allows agents to discover optimal policies through trial and error and is well-suited for complex, long-horizon behaviors. For instance, the Skill Discovery Framework from UC Berkeley [41] extracts high-value action primitives through unsupervised interaction, while Google's Learning from Play project [42] mines latent behavioral structures from unstructured human play data.

As predicate and action learning methods mature, increasing emphasis is placed on self-reflection, automatic parameter tuning, and minimizing manual intervention. A key challenge remains in how to flexibly compose and synthesize new action primitives to improve task efficiency and enable practical deployment of intelligent robots in industrial environments.

3. Overview of the Proposed System

As illustrated in Figure 1, the proposed NeuroSymbolic disassembly system for electric vehicle battery (EVB) packs integrates perception, decision making, and execution into a unified architecture. This section presents a brief overview of the system from two perspectives: (i) hardware composition and (ii) the core functional modules for perception, planning, and control.

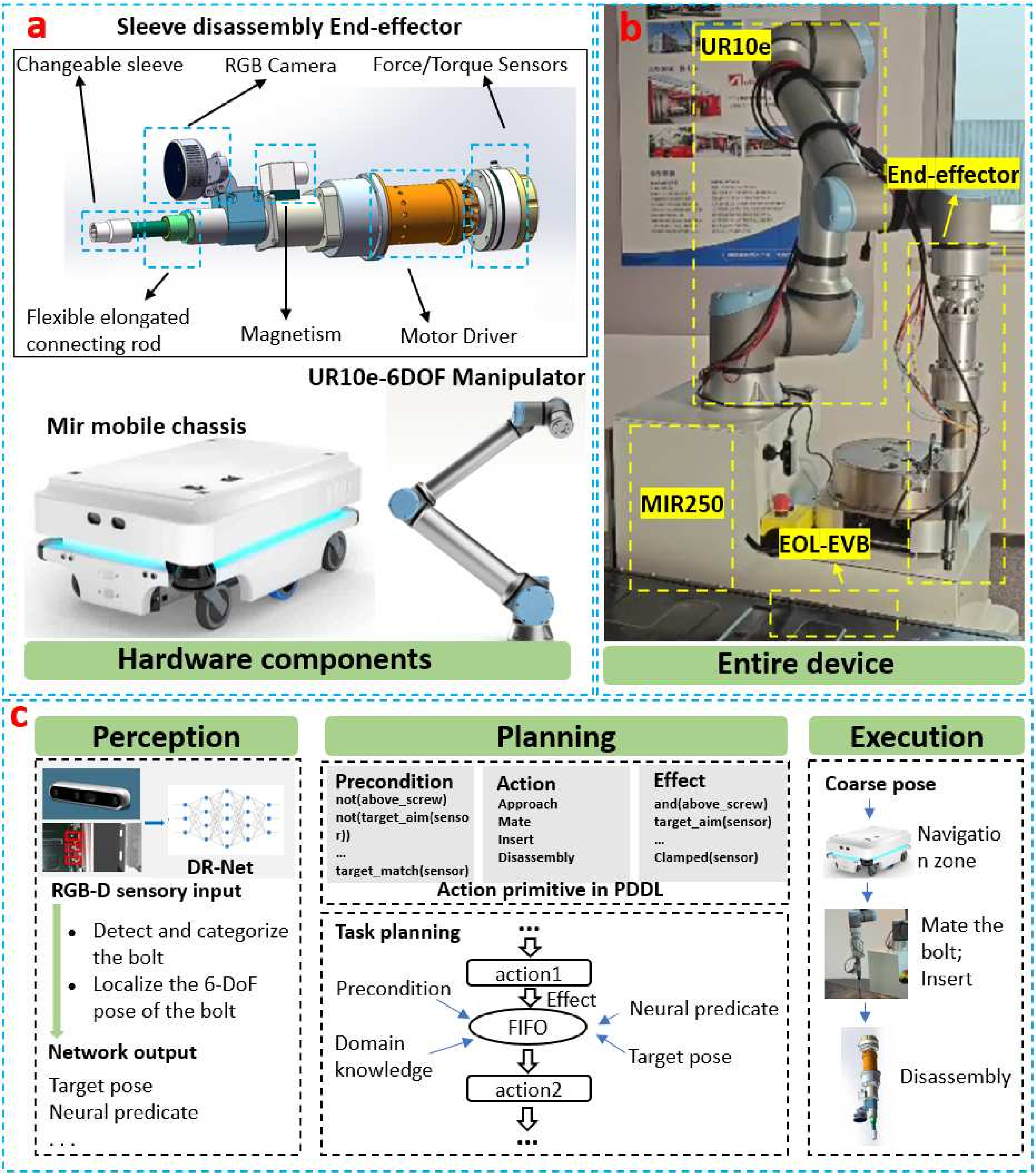

Figure 1. System architecture of the proposed NeuroSymbolic disassembly platform for electric vehicle battery packs. (**a**) Hardware components, including the MiR250 autonomous mobile robot, the UR10e collaborative manipulator, and the custom-designed bolt disassembly end-effector. (**b**) Full physical setup of the integrated platform. (**c**) Functional modules of the system, illustrating the perception, planning, and execution pipelines enabling autonomous operation.

3.1. System Hardware Architecture

The hardware of the embodied intelligent disassembly system consists of three primary components: an autonomous mobile robot, a six-degree-of-freedom robotic manipulator, and a modular sleeve-type end-effector.

3.1.1. Autonomous Mobile Robot

We employ the MiR250 autonomous mobile robot (Mobile Industrial Robots A/S, Odense, Denmark), which features a compact design, high payload capacity, robust navigation capabilities, and seamless integration with collaborative manipulators. The AMR provides global mobility and positioning for the disassembly platform, allowing the robot to access various fastening locations and overcoming the reachability limitations of fixed-base robotic arms.

3.1.2. Six-Degree-of-Freedom Manipulator

The Universal Robots UR10e collaborative manipulator (Universal Robots A/S, Odense, Denmark) is selected for its ease of programming, rapid deployment, and built-in safety features, making it well-suited for human–robot collaboration. Widely adopted in material handling and manufacturing, the UR10e supports a standardized flange interface that enables quick interchangeability of end-effectors.

3.1.3. Modular Sleeve-Type End-Effector

The end-effector integrates both perception and execution modules:

- **Perception Module:**
  - *Vision Sensing:* An Intel RealSense RGB-D camera captures high-resolution depth and color data for visual analysis.
  - *Force/Torque Sensing:* An ATI six-axis force/torque sensor(ATI Industrial Automation, Inc., Apex, NC, USA) continuously monitors disassembly forces and moments, enabling safe and precise interaction with components.
- **Execution Module:**
  - *Modular Sleeve Adapter:* A flexible connector for switching screwdriver sockets to accommodate various bolt specifications.
  - *Drive Motor:* It delivers sufficient torque for reliable bolt loosening and removal.
  - *Electromagnet:* It enables magnetic capture and retention of removed bolts through electrically actuated absorption.
  - *Flexible Joint:* It compensates for surface irregularities commonly found on aged battery packs, ensuring stable and consistent contact with fasteners.

3.1.4. Experimental Platform

All experiments presented in this paper are conducted on the described hardware platform, where the complete NeuroSymbolic self-learning disassembly framework is deployed and evaluated.

3.2. System Fundamentals

The proposed NeuroSymbolic robotic disassembly framework incorporates neural predicates and action primitives to achieve autonomous, interpretable, and adaptive behavior in complex EVB disassembly tasks.

Neural predicates are neural networks that map continuous sensor inputs into discrete symbolic states. They serve as a critical bridge between low-level perception, such as RGB-D data and force/torque signals, and high-level symbolic reasoning. By processing data from multimodal sensors, neural predicates extract semantically meaningful features and represent them within a symbolic state space. Unlike static, manually defined abstractions, neural predicates enable dynamic adaptation to unstructured environments, significantly enhancing system robustness and decision-making autonomy.

Action primitives decompose the disassembly task into atomic, non-divisible operations based on domain knowledge and expert demonstrations. Inspired by industrial disassembly procedures, we define a set of essential action primitives, listed in Table 1, to cover operations such as alignment, insertion, and unscrewing. Each primitive is formally specified using the PDDL, which defines its symbolic preconditions and expected outcomes. Execution of these primitives is triggered by symbolic states inferred from neural predicates, enabling precise and context-aware task planning.

Table 1. Core action primitives for EV battery bolt disassembly.

Action Primitive	Function
Approach	The robotic end-effector moves toward the nearest detected bolt within the visual field.
Mate	The bolt position is refined, and the end-effector is aligned accurately along the bolt axis.
Insert	The socket tool is lowered to securely engage the bolt head.
Disassemble	The end-effector applies a counterclockwise torque to loosen and remove the target bolt.

Within this knowledge-driven framework, neural predicates and action primitives operate in a tightly coupled loop: predicates interpret raw sensor data into symbolic states, which then guide the symbolic planner to sequence and trigger suitable action primitives that meet the PDDL-defined criteria. This NeuroSymbolic integration enables the system to achieve autonomous perception, planning, fine-grained execution, and continual adaptation, laying the groundwork for scalable, explainable robotic disassembly in real-world industrial applications.

4. Method

While previous NeuroSymbolic frameworks for autonomous electric vehicle (EV) battery disassembly have demonstrated accurate and reliable screw removal capabilities, they still lack efficient and self-improving learning mechanisms. In particular, these systems are unable to adaptively refine disassembly strategies across repeated tasks, resulting in redundant or suboptimal actions and ultimately limiting their scalability toward industrial-grade embodied intelligence.

To address the above issues, we propose an enhanced framework, as illustrated in Figure 2, which integrates LLMs and autonomous pattern learning algorithms into the existing NeuroSymbolic architecture. The upgraded system optimizes task strategies based on historical disassembly experience, autonomously learns new action primitives, and iteratively refines neural predicates to achieve more efficient task execution. The overall framework consists of two complementary modules: an offline LLM inference module that analyzes updated historical databases to determine potential optimizations and an embedded deployment module for action primitive optimization and neural predicate training, which integrates learned knowledge directly into the robot's disassembly operations. This architecture significantly enhances the autonomous battery disassembly system's capabilities in terms of task understanding, strategy generation, and execution efficiency.

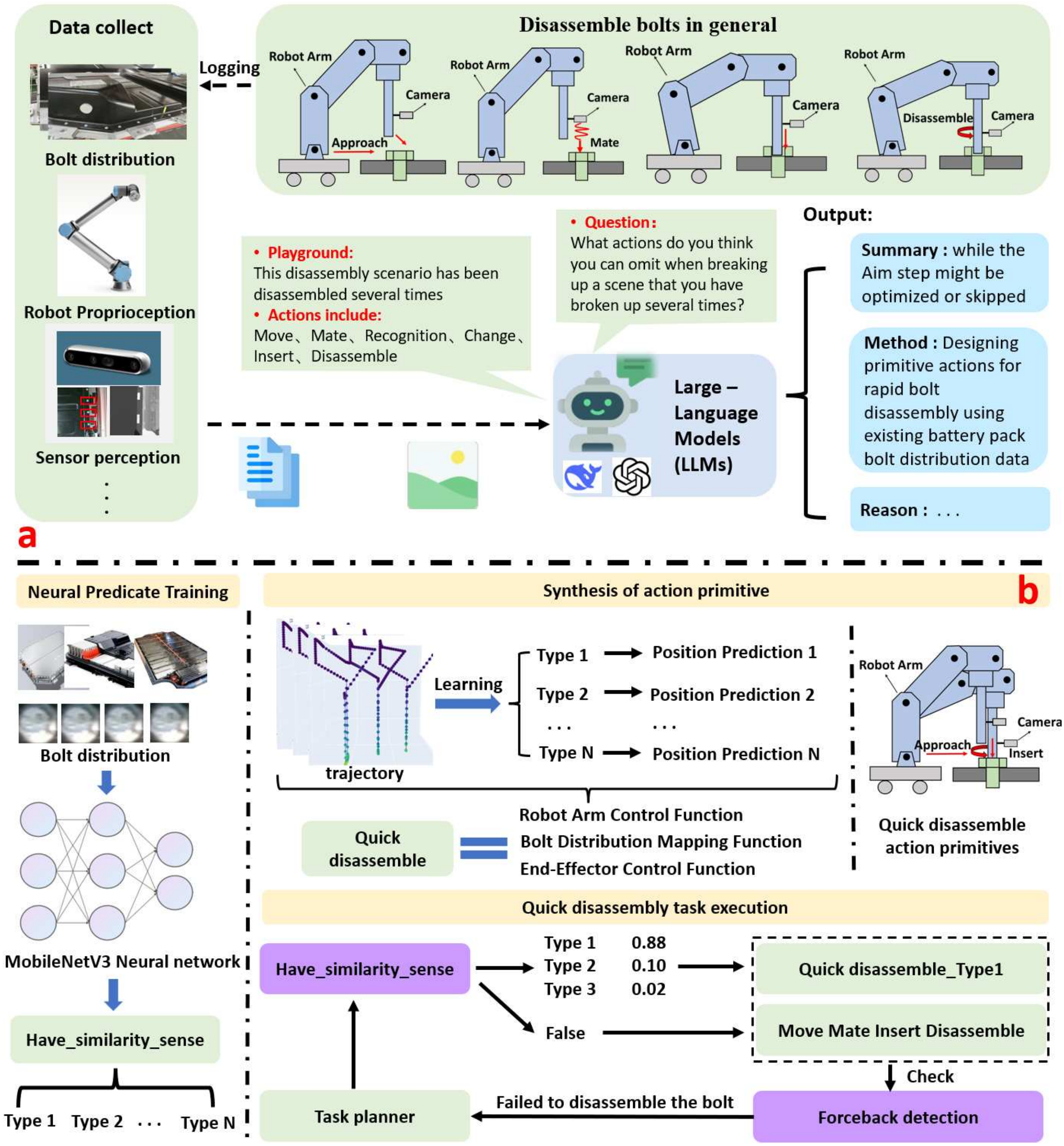

Figure 2. Overview of the autonomous learning framework for efficient bolt disassembly. (**a**) Information acquisition and LLM-based planning module, encompassing real-time data collection during disassembly and data-driven analysis using LLMs. (**b**) Neural predicate and action primitive execution module, illustrating the training of new neural predicates, the learning and synthesis of efficient action primitives, and the integration of feedback-based verification during execution.

4.1. Bolt Disassembly Data Collection and Storage

To support experience-driven pattern discovery and policy optimization, we design a structured pipeline to collect key spatial information during the bolt disassembly process of EV battery packs. Compared with conventional time-series trajectory data, this pipeline emphasizes the spatial distribution of disassembly primitives and their relationship to bolt positions in a consistent reference frame.

4.1.1. Data Definitions and Mathematical Representation

- **Battery Identifier:** Let $\mathcal{B}$ denote the set of battery models, with each individual instance being represented as $b \in \mathcal{B}$.
- **Bolt Position Set:** For a given battery b, the set of all bolt positions is defined as

$$P_b = \left\{ \mathbf{p}_k \in \mathbb{R}^3 \mid k = 1, \ldots, n_b \right\},$$

  where n_b is the number of bolts and each $\mathbf{p}_k = (x_k, y_k, z_k)$ represents the 3D position of bolt k in a fixed global coordinate frame.
- **Action Primitive Set:** For each bolt disassembly action, the corresponding action primitives are recorded as

$$A_b = \{(a_k, \mathbf{p}_k^s, \mathbf{p}_k^e) \mid k = 1, \ldots, n_b\},$$

  where
  - a_k denotes the action primitive label (e.g., `move`, `insert`, `mate`);
  - $\mathbf{p}_k^s \in \mathbb{R}^3$ is the start position of the end-effector before executing a_k;
  - $\mathbf{p}_k^e \in \mathbb{R}^3$ is the end position after execution.

4.1.2. Storage Format and System Implementation

Each disassembly session is represented as

$$e_b = (b, P_b, A_b, t_{\text{start}}),$$

where t_{start} denotes the start time of the session. As this design focuses on spatial reasoning, joint-space trajectories are intentionally excluded.

The data are stored using a hierarchical HDF5 or JSON-based format, exemplified as follows:

```
/session_<ID>/
  metadata/
    battery_id: ‘‘BATT-001’’
    bolt_count: 8
    bolt_positions: [[x1,y1,z1], ..., [x8,y8,z8]]
  actions/
    [
      {‘‘primitive’’: ‘‘move’’,
       ‘‘start_position’’: [x, y, z],
       ‘‘end_position’’: [x, y, z]},
       ...
    ]
```

The above-described data collection module provides a structured and automated mechanism for capturing critical information during each bolt disassembly session, including the battery model, bolt positions, and the start and end poses of the end-effector for each action. By maintaining a consistent spatial reference frame and standardized schema, the system supports reliable accumulation of experiential data across diverse disassembly scenarios.

Although the current focus is on data acquisition, the collected information lays a robust foundation for a range of future developments. These include but are not limited to statistical analysis of bolt layouts, rule-based or learned action sequence optimization, and integration with machine learning or reinforcement learning frameworks for adaptive policy generation.

4.2. LLM-Driven Contextual Reasoning and Strategy Optimization

While the previously introduced NeuroSymbolic disassembly framework enables autonomous bolt removal for electric vehicle (EV) battery packs, it lacks the capability to abstract and reuse past experiences. As a result, the system must reinitiate disassembly planning from scratch for each task, leading to inefficiencies. To address this limitation, we introduce an LLM as a contextual reasoning engine. The LLM analyzes the current disassembly scenario and consults a historical experience database to suggest optimized action sequences or propose new neural predicates and action primitives. This subsection outlines the design and integration of the LLM into our disassembly framework. The overall architecture is illustrated in Figure 3, which shows how the LLM leverages prompt design to drive task optimization.

As a dedicated reasoning module, the LLM interprets the current situation based on multiple contextual inputs, including the battery pack model, spatial distribution of fasteners, and task-specific execution history (with a current focus on outer shell bolt removal and future extensibility to module-level operations). Because the LLM does not receive explicit task annotations, prompt engineering plays a critical role in improving its semantic understanding and planning quality [43–45]. To this end, we design five structured prompts to guide the LLM in reasoning about disassembly optimization:

1. **Role Definition:** "You are a NeuroSymbolic battery disassembly robot equipped with autonomous decision-making capabilities. Your primary task is to remove bolts from battery packs. You are familiar with the standard disassembly process."
2. **Historical Case Examples:**
   Example 1
   ```
   battery_id: A
   bolt_positions: [[...]]
   task_planner: [[approach, mate, recognition, insert, disassemble], ...]
   ```
3. **Analogy Reminder:** Humans often refine their actions through repetition. For example, after repeatedly entering the same room to switch on a light, they gradually discover more efficient and safer routes. The time required for this task decreases with each repetition due to accumulated experience and behavior optimization.
4. **Task-Specific Prompt:** "Based on your current disassembly task, analyze whether there is a regularity in the end-effector's motion trajectories. Can these patterns be exploited to generate optimized trajectories? Can new action primitives be synthesized to execute the task more efficiently based on learned patterns?"
5. **Output Format Constraint:** "Please respond using the following JSON format:
   ```
   {
   motion_pattern: true,
   optimization_possible: true,
   suggestions: ‘‘<Textual optimization suggestions>’’,
   reasoning: ‘‘<Explanation of why the optimization is beneficial>’’
   }
   ```

To balance reasoning accuracy and real-time performance, we employ the `DeepSeek-R1` model as the primary LLM. Additionally, distilled and quantized 7B and 8B variants are deployed for on-device inference acceleration. To further assess model generalizability, we also perform tests with `ChatGPT-4o` and `GPT-4o-mini` as comparative baselines.

By combining retrieval-augmented prompts, structured output constraints, and scalable deployment strategies, this LLM module enables the disassembly system to generalize across different battery configurations, reuse prior experience, and dynamically adapt its planning behavior in real time. Consequently, it forms a robust foundation for intelligent, efficient, and scalable robotic disassembly of EV battery packs.

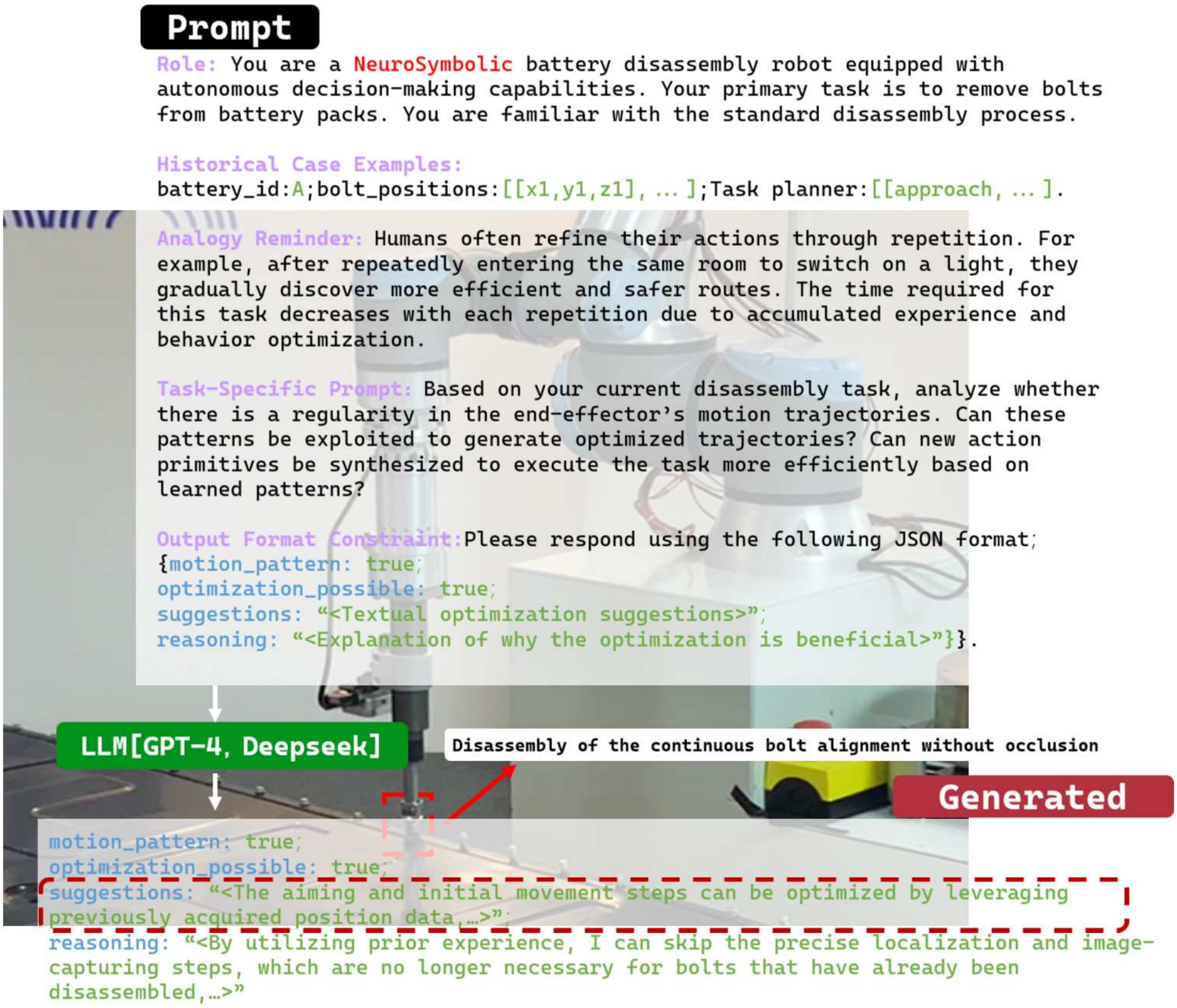

Figure 3. LLM-driven reasoning architecture for task optimization in scenario of continuous bolt alignment without occlusion. The diagram illustrates our prompt design and an example output from the LLM, showcasing how prompt-based inference is used to streamline motion planning and reduce redundant actions.

4.3. Bolt Position Prediction Based on Geometric Priors

Building on the optimization suggestions generated by the LLM, our system requires the capability to autonomously and efficiently identify spatial regularities in the disassembly environment. This enables the synthesis of novel and efficient action primitives that are well-aligned with the detected geometric patterns. However, given that LLMs are based on neural architectures and may occasionally generate hallucinated outputs, we incorporate a bolt distribution consistency detection module to enhance robustness and ensure accurate pattern recognition.

Let the detected bolt positions be represented as $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n\}$, where each $\mathbf{p}_i \in \mathbb{R}^3$ denotes the 3D coordinate of the i-th bolt. To evaluate spatial consistency, we compute the Euclidean distance between each pair of adjacent bolts:

$$d_i = \|\mathbf{p}_{i+1} - \mathbf{p}_i\|_2, \quad i = 1, 2, \ldots, n-1 \tag{1}$$

This produces a set of spacing distances $\mathcal{D} = \{d_1, d_2, \ldots, d_{n-1}\}$. Two statistical measures are employed to quantify the spatial regularity:

- **Standard deviation** σ_d, which reflects global variance:

$$\bar{d} = \frac{1}{n-1}\sum_{i=1}^{n-1} d_i, \quad \sigma_d = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n-1}(d_i - \bar{d})^2} \tag{2}$$

- **Maximum spacing deviation** Δ_d, which captures the range of bolt spacing:

$$\Delta_d = \max(\mathcal{D}) - \min(\mathcal{D}) \tag{3}$$

To determine whether a given bolt configuration exhibits approximate regularity, we define a threshold ϵ. If $\sigma_d \leq \epsilon$ or $\Delta_d \leq \epsilon$, the bolt pattern is classified as approximately equidistant, implying a regular structural layout.

This consistency assessment serves as a key prior for downstream tasks such as geometric fitting and bolt position prediction, supporting the reuse of optimized action primitives in structured disassembly settings.

In many disassembly tasks, bolts are arranged in geometrically consistent patterns, particularly along battery enclosures or motor housings, where they exhibit uniform spacing along linear or curved paths. To exploit this prior knowledge, we introduce a baseline prediction approach that leverages average spacing and directional alignment.

Let the known bolt positions be defined as $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n\}$, with each $\mathbf{p}_i \in \mathbb{R}^3$ representing the 3D coordinates of bolt i. The Euclidean distance between consecutive bolts is

$$d_i = \|\mathbf{p}_{i+1} - \mathbf{p}_i\|_2, \quad \text{for } i = 1, 2, \ldots, n-1 \tag{4}$$

The average spacing $\bar{d}$ is computed as

$$\bar{d} = \frac{1}{n-1}\sum_{i=1}^{n-1} d_i \tag{5}$$

To estimate the dominant alignment direction, we perform Principal Component Analysis (PCA) on the set $\mathcal{P}$ and extract the principal direction vector $\hat{\mathbf{v}}$. Based on this, we define a geometric prediction function $f(k)$ that estimates the position of the k-th bolt, assuming that the first bolt $\mathbf{p}_1$ is known:

$$f(k) = \mathbf{p}_1 + (k-1) \cdot \bar{d} \cdot \hat{\mathbf{v}}, \quad \text{for } k = 1, 2, \ldots, m \tag{6}$$

This approach provides accurate bolt position predictions in scenarios with evenly spaced bolts and a consistent alignment direction. It enables the system to infer the positions of unobserved bolts using only a few initial detections, thereby reducing the perception and computation burden. As a lightweight geometric baseline, this method serves as a reference framework for more sophisticated generalization models.

To further enhance robustness, we integrate a force feedback verification mechanism [8] to validate predicted bolt positions. After estimating position $\hat{\mathbf{p}}_k$ by using $f(k)$, the robot executes a standard insertion primitive A_k. Following the action, a force/torque sensor (an ATI six-axis sensor) collects wrench data $\mathbf{F}_k(t)$ from the end-effector. A pre-calibrated threshold set $\mathcal{F}_{\text{success}}$ determines insertion success:

$$C_k = \mathbb{I}(\mathbf{F}_k(t) \in \mathcal{F}_{\text{success}}) \tag{7}$$

Here, $C_k = 1$ indicates successful insertion, while $C_k = 0$ triggers a fallback mechanism. In case of failure, the fast disassembly routine is interrupted, and the system reverts to conventional perception and planning to re-estimate the bolt position $\hat{\mathbf{p}}_k^{\text{new}}$, thereby ensuring safety and precision.

This closed-loop mechanism enhances the robustness of predictive manipulation in complex environments. It allows real-time correction in response to environmental variation or model uncertainty, effectively preventing operational errors and collisions.

In summary, the bolt position prediction and execution module operates through a three-stage closed-loop process:

- **Spacing Consistency Detection:** Adjacent bolt positions are used to compute spacing distances, and statistical measures such as standard deviation and spacing range are applied to assess layout regularity.
- **Bolt Position Prediction:** Given a regular pattern, average spacing and principal direction are used to build a predictive function $f(k)$, allowing for the extrapolation of bolt positions from a known reference point.
- **Force Feedback Verification and Correction:** After performing the primitive insertion action, real-time force data are analyzed. In the event of a failed insertion, the system halts execution and switches to conventional planning for correction.

This closed-loop pipeline integrates perception, prediction, and verification, enhancing operational efficiency while ensuring high robustness and fault tolerance. It is particularly well-suited for automated bolt disassembly tasks involving repetitive structural patterns.

4.4. Training of the Similar Scene Recognition Predicate

As discussed in previous sections, our system leverages LLM-based reasoning and historical data to learn predictive models for bolt positions in familiar scenarios, as well as novel action primitives for rapid disassembly. During task execution, however, it becomes critical to assess whether the current environment resembles a previously encountered scene, in order to trigger these learned primitives. To address this, we introduce a neural predicate, `Have_similarity_scene`, designed to evaluate scene similarity and determine whether an accelerated disassembly strategy should be activated.

The predicate is implemented using MobileNetV3 [46,47], a lightweight yet high-performance convolutional neural network tailored for embedded and real-time applications. Compared with other backbone networks such as AlexNet, InceptionV3 [48], and ShuffleNetV2 [49], MobileNetV3 offers a superior balance between accuracy and computational efficiency. It integrates depthwise separable convolutions with neural architecture search (NAS) techniques to jointly optimize model performance and latency, making it especially suitable for robotic perception tasks under constrained resources.

To train the MobileNetV3 model, we collected high-resolution RGB-D data from four representative EV battery disassembly scenarios using an Intel RealSense depth camera. Each scenario contains 200 RGB images paired with 200 corresponding depth maps, capturing both visual and geometric features. To enhance model generalization under real-world variability, we applied extensive data augmentation. Specifically, each image underwent five occlusion transformations and eight rotational augmentations, expanding the dataset by a factor of 80. This yielded a final dataset of 64,000 images (RGB + depth), offering high visual diversity for robust feature learning.

The dataset covers the following four scene categories, as shown in Figure 4:

- Continuous bolt alignment without occlusion;
- Continuous bolt alignment with partial occlusion;
- Intermittent bolt alignment without occlusion;
- Intermittent bolt alignment with partial occlusion.

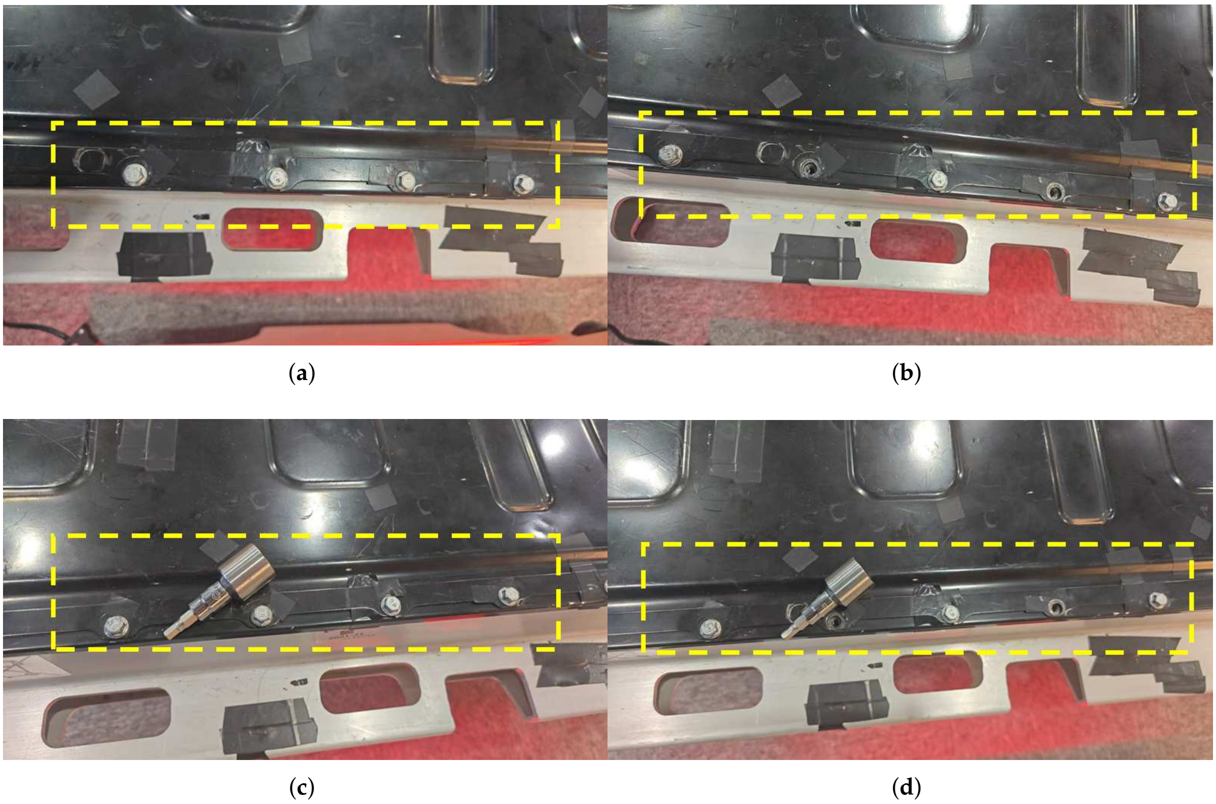

Figure 4. Four scenarios of bolt distribution in automotive power battery packs: (**a**) Continuous bolt distribution without obstacles. (**b**) Discontinuous bolt distribution without obstacles. (**c**) Continuous bolt distribution with obstacles. (**d**) Discontinuous bolt distribution with obstacles. In each scenario, the bolt distribution is highlighted using yellow dashed squares.

All samples were standardized and manually annotated to ensure high-quality supervision. During inference, MobileNetV3 performs real-time feature extraction from depth images and evaluates whether the current scene matches any previously encountered ones based on learned embeddings. If a match is detected, the `Have_similarity_scene` predicate is set to `True`, thereby triggering the pre-optimized task plan and activating the bolt position prediction module. This mechanism bypasses redundant perception processes and improves execution efficiency.

As shown in Algorithm 1, the neural predicate provides a reliable mechanism for scene classification. When a similar scene is identified, the system initiates a fast-track disassembly procedure using previously learned strategies, significantly improving adaptability and operational efficiency in repetitive disassembly tasks. This design greatly enhances the intelligence and practicality of the robotic system.

Algorithm 1 Scene classification and action execution via neural predicate.

```
function classify_scene_and_act(image_path)
    # Step 1: Load the pre-trained MobileNetV3 classifier
    model ← load_scene_classifier(MobileNetV3)

    # Step 2: Classify the input image to obtain scene label
    scene_label ← model.classify(image_path)

    # Step 3: Activate the corresponding predicate
    Activate_predicate(scene_label)

    # Step 4: Retrieve and apply corresponding action mapping
    scene_mapping ← get_scene_mapping(scene_label)
    apply_mapping(scene_mapping)

    # Step 5: Execute fast disassembly action
    quick_disassemble()
    return True
end function

result ← classify_scene_and_act(image)
print("Scene label and action triggered:", result)
```

5. Experiments

We evaluate the proposed self-learning framework on the task of bolt disassembly from electric vehicle battery (EVB) packs. To assess the effectiveness of LLM-based strategy optimization, as well as the performance of the newly learned neural predicates and action primitives in facilitating efficient and continuous disassembly, we design a series of controlled comparative experiments. The experiments are structured in two phases:

- **Baseline Method:** A conventional NeuroSymbolic system without optimization, which performs continuous bolt disassembly by using static, predefined strategies.
- **Optimized Method:** The proposed self-learning-enhanced NeuroSymbolic system, which integrates LLM-driven strategy adaptation and autonomously learned primitives to perform the same disassembly tasks.

To demonstrate the generalizability and robustness of the proposed approach, we conduct real-world experiments on an actual EVB in a laboratory environment. Four distinct scene conditions are designed to simulate typical structural and occlusion scenarios encountered during battery disassembly. These scenarios represent

1. Uniform bolt distribution without occlusion;
2. Uniform bolt distribution with partial occlusion;
3. Irregular bolt alignment without occlusion;
4. Irregular bolt alignment with partial occlusion.

For each scene, both methods are evaluated based on two key metrics: total disassembly time and operational efficiency. By comparing performance across all conditions, we demonstrate that the optimized method consistently outperforms the baseline.

The experimental results show that the enhanced NeuroSymbolic system, empowered by LLM-based contextual reasoning and continuous learning, can adapt its disassembly strategy dynamically based on prior experience. This leads to significant improvements in task efficiency and highlights the system's potential for achieving higher levels of autonomy and adaptability in real-world industrial disassembly tasks.

5.1. LLM-Based Optimization Mechanism Evaluation

This study investigates the adaptability of LLMs to bolt disassembly tasks, focusing on their ability to generate effective optimization recommendations. We evaluate representative LLMs—namely, GPT-4o-mini, GPT-4o, and DeepSeek-R1 models (7B and 8B parameters)—integrated into our experimental framework via API interfaces. These models have demonstrated strong language comprehension and reasoning capabilities, making them well-suited for complex task planning in robotic systems.

5.1.1. Experimental Setup

To ensure a rigorous and comprehensive evaluation, we adopt a controlled experimental design that categorizes test scenarios based on the spatial arrangement of bolts and the presence of physical obstructions:

- Continuous bolt distribution without obstacles;
- Discontinuous bolt distribution without obstacles;
- Continuous bolt distribution with obstacles;
- Discontinuous bolt distribution with obstacles.

To maintain consistency across all experiments, we standardize the prompt structure, including the playground and example sections. Only the task-specific question component is modified to reflect variations in scene complexity. Each model is tested over 50 independent trials per scenario. Accuracy and performance are assessed using statistical methods to quantify the feasibility of employing LLMs as optimization engines in autonomous disassembly workflows.

5.1.2. Results

The performance evaluation in this experiment was conducted along three key dimensions: task comprehension ability, the quality of optimization suggestions, and the practical feasibility of those suggestions. Scores were assigned based on the responses generated by various LLMs. As illustrated in Figure 5, the results demonstrate that current mainstream LLMs exhibit significant application potential in the context of robotic disassembly optimization.

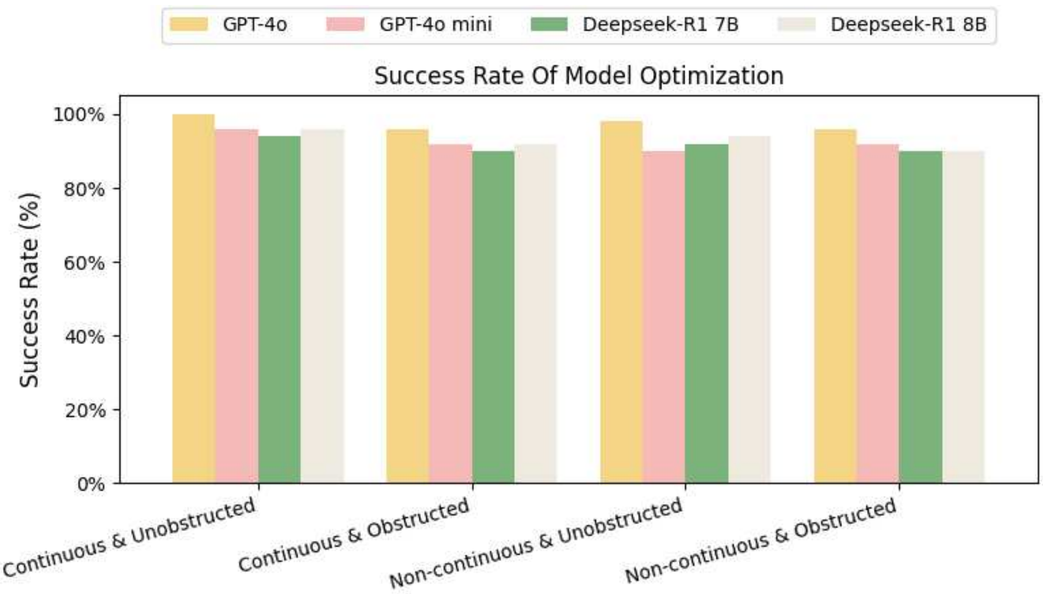

Figure 5. Experimental results of LLM-based optimization across four disassembly scenarios.

In the simplest case of *continuous bolt distribution without occlusion*, all evaluated models achieved a 100% success rate in generating valid optimization strategies. This indicates that LLMs are capable of leveraging scene understanding and historical experience to infer whether a disassembly task can be simplified and can generate efficient action plans without relying on pre-defined PDDL task descriptions.

For more challenging scenarios, such as *discontinuous bolt layouts* or *occluded bolts*, all tested models still maintained over 94% accuracy. This further confirms the generalization capabilities of LLMs in complex and dynamic environments. When equipped with well-designed prompts, LLMs can reliably support task-specific optimization across diverse settings.

Notably, smaller-scale models with 7B and 8B parameters performed competitively in terms of consistency and interpretability of the suggestions. This suggests that effective deployment in industrial applications does not necessarily require large-scale models, thus significantly reducing computational costs and deployment barriers.

In summary, the experimental results provide strong empirical evidence for integrating LLMs into industrial automation workflows. The findings also highlight future directions: prompt engineering remains a critical factor for performance, and expanding the scope toward multimodal visual-language models (VLMs) [50] holds promise for further enhancing adaptability in more complex and dynamic EV battery disassembly environments.

5.2. Bolt Disassembly Experiments

To evaluate the feasibility and accuracy of the newly generated neural predicates and action primitives derived from LLM-based optimization suggestions, we conducted a set of robot-assisted bolt disassembly experiments. The test scenarios involved M8 hex-head bolts arranged in continuous and unoccluded configurations, with variations including two, three, and four consecutively aligned bolts. These were compared against a traditional method involving sequential removal of bolts one by one. Task performance was quantified based on bolt disassembly success rate and total execution time, aiming to validate the effectiveness and practicality of LLM-generated optimization strategies in real robotic operations. Through the design of the above bolt alignment scenarios, we further assessed the generalization and adaptability of the method under different simulated EV battery configurations.

5.2.1. Experimental Setup

The experimental platform utilized a UR10e robotic arm equipped with a vision sensor, a force/torque sensor, interchangeable socket tools, and a high-speed DC motor at the end-effector (see Figure 1a). This setup enabled automated removal of structural bolts from EV battery packs. Ambient lighting was carefully controlled to eliminate glare and ensure reliable visual input. For each bolt configuration, 20 trials were performed, and both the number of successfully removed bolts and task completion time were recorded for performance evaluation.

5.2.2. Result

The results indicate that the neural predicates and action primitives generated from autonomous learning significantly improved operational efficiency. In the three-bolt alignment scenario, a total of 180 bolts were removed across 20 trials, achieving a 97.8% success rate with an average execution time of just 1 min and 42 s. In contrast, the traditional sequential method maintained a 100% success rate but required an average of 4 min and 17 s per trial. This represents a 154.4% improvement in disassembly efficiency. Additional results are summarized in Table 2.

Error analysis revealed that occasional failures in the optimized approach stemmed from the accumulation of minor positional errors during sequential actions. Although the new action primitives were able to infer bolt positions from learned spatial patterns, these incremental deviations became more pronounced in scenarios with a larger number of bolts, sometimes leading to missed disassemblies. Future work will incorporate correction mechanisms to mitigate such errors and improve reliability in longer sequences.

Overall, the proposed NeuroSymbolic autonomous learning framework achieves both high task success rates and improved execution efficiency in repetitive disassembly scenarios. These findings also highlight the practical potential of integrating LLMs into robotic task planning and adaptive control.

Table 2. Success rate and disassembly time for different tasks.

Disassembly Task	Successful Attempts	Unsuccessful Attempts
Quick disassembly of 3 bolts (total: 9)	173	7
Single disassembly of 9 bolts (total: 9)	179	1
Quick disassembly of 2 bolts (total: 8)	156	4
Quick disassembly of 4 bolts (total: 8)	152	8
Single disassembly of 8 bolts (total: 8)	180	0
Disassembly Task	**Success Rate**	**Average Time**
Quick disassembly of 3 bolts (total: 9)	96.10%	1 min 41 s
Single disassembly of 9 bolts (total: 9)	99.40%	4 min 17 s
Quick disassembly of 2 bolts (total: 8)	97.50%	1 min 38 s
Quick disassembly of 4 bolts (total: 8)	95.00%	1 min 21 s
Single disassembly of 8 bolts (total: 8)	100.00%	3 min 45 s

6. Discussion

To address the limitations of conventional NeuroSymbolic disassembly systems in leveraging historical experience for efficient bolt removal in electric vehicle battery (EVB) packs, this study proposes a self-learning framework that derives optimized disassembly strategies and action primitives from historical task data. By incorporating an LLM as the system's core reasoning engine, our framework analyzes both task history and scene context to determine the potential for simplification and optimization. A trajectory learning algorithm is also introduced to extract key positions from previously optimized trajectories, allowing the system to bypass redundant visual recognition steps in repetitive tasks.

Experimental evaluations on a real-world robotic disassembly platform demonstrate that the proposed method significantly improves disassembly performance. Specifically, the average time to remove each bolt is reduced by 17 s, corresponding to a 154.4% improvement in overall disassembly efficiency. These results validate the effectiveness of leveraging historical task experience to derive efficient and reusable bolt disassembly strategies.

However, several limitations remain: the current LLM operates offline and is not yet integrated into the system for real-time interaction; the diversity of EVB models in the dataset is still limited; and the trajectory learning algorithm is primarily applicable to surface bolts, lacking generalization to more irregular or hidden fasteners.

Future work will focus on collecting larger-scale datasets across a wider range of EVB models to enhance generalizability. We also plan to integrate reinforcement learning, imitation learning, and knowledge distillation techniques to learn more robust and generalized disassembly strategies, including those for internal modules. Moreover, we will improve LLM prompt engineering to support more complex scenarios and investigate its capabilities in code generation and error correction. Ultimately, we aim to build a fully autonomous and adaptive battery disassembly system capable of real-time understanding, strategy refinement, and self-correction across diverse and uncertain environments.

7. Conclusions

This study proposes a NeuroSymbolic self-learning framework that integrates an LLM and a trajectory learning algorithm to enhance the efficiency of electric vehicle battery pack bolt disassembly. The method overcomes the limitations of relying solely on predefined PDDL, enabling action optimization in familiar scenarios. Experimental results show that the framework reduces the average bolt removal time by 17 s and improves overall efficiency by 154.4%, validating the effectiveness of leveraging historical task experience for reusable disassembly strategies. This work provides a feasible solution for intelligent battery disassembly and expands the self-optimization capability of NeuroSymbolic approaches.

Author Contributions: Data curation: P.C., Z.H., and Y.P.; formal analysis: Z.W. and M.C.; funding acquisition: Z.W. and M.C.; investigation: P.C. and Z.W.; methodology: P.C. and Z.W.; project administration: M.C. and Z.W.; resources: P.C. and Y.P.; software: P.C.; supervision: P.C. and Y.P.; validation: P.C. and Z.H.; visualization: P.C. and Z.H.; writing—original draft P.C.; writing—review and editing: P.C., Z.W., and M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Industry and Information Technology of China within the program "2021 High Quality Development Project (TC210H02C)".

Data Availability Statement: The original contributions presented in this study are included in the article. The datasets generated and analyzed during the current study are available from the corresponding author upon reasonable request.

Acknowledgments: The authors sincerely thank the editors and anonymous reviewers for their thorough review and constructive suggestions, which have greatly improved the quality of this manuscript. We are also grateful to all colleagues and collaborators who contributed to this research, as well as the supporting institutions and laboratories that facilitated this work.

Conflicts of Interest: Author Zhigang Wang was employed by the company Intel Labs China. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Beaudet, A.; Larouche, F.; Amouzegar, K.; Bouchard, P.; Zaghib, K. Key Challenges and Opportunities for Recycling Electric Vehicle Battery Materials. *Sustainability* **2020**, *12*, 5837. [CrossRef]
2. Xiao, J.; Jiang, C.; Wang, B. A review on dynamic recycling of electric vehicle battery: disassembly and echelon utilization. *Batteries* **2023**, *9*, 57. [CrossRef]
3. Yun, L.; Linh, D.; Shui, L.; Peng, X.; Garg, A.; LE, M.L.P.; Asghari, S.; Sandoval, J. Metallurgical and mechanical methods for recycling of lithium-ion battery pack for electric vehicles. *Resour. Conserv. Recycl.* **2018**, *136*, 198–208. [CrossRef]
4. Weyrich, M.; Wang, Y. Architecture design of a vision-based intelligent system for automated disassembly of E-waste with a case study of traction batteries. In Proceedings of the 2013 IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA), Cagliari, Italy, 10–13 September 2013 ; IEEE: Piscataway, NJ, USA, 2013; pp. 1–8.
5. Hellmuth, J.F.; DiFilippo, N.M.; Jouaneh, M.K. Assessment of the automation potential of electric vehicle battery disassembly. *J. Manuf. Syst.* **2021**, *59*, 398–412. [CrossRef]
6. Zhang, H.; Yang, H.; Wang, H.; Wang, Z.; Zhang, S.; Chen, M. Autonomous electric vehicle battery disassembly based on neurosymbolic computing. In Proceedings of the SAI Intelligent Systems Conference, Amsterdam, The Netherlands, 1–2 September 2022; Springer: Berlin/Heidelberg, Germany, 2022, pp. 443–457.
7. Zhang, H.; Zhang, Y.; Wang, Z.; Zhang, S.; Li, H.; Chen, M. A novel knowledge-driven flexible human–robot hybrid disassembly line and its key technologies for electric vehicle batteries. *J. Manuf. Syst.* **2023**, *68*, 338–353. [CrossRef]
8. Zhang, Y.; Zhang, H.; Wang, Z.; Zhang, S.; Li, H.; Chen, M. Development of an autonomous, explainable, robust robotic system for electric vehicle battery disassembly. In Proceedings of the 2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Cagliari, Italy, 10–13 September 2013; IEEE: Piscataway, NJ, USA, 2023, pp. 409–414.

9. Peng, Y.; Wang, Z.; Zhang, Y.; Zhang, S.; Cai, N.; Wu, F.; Chen, M. Revolutionizing Battery Disassembly: The Design and Implementation of a Battery Disassembly Autonomous Mobile Manipulator Robot (BEAM-1). In Proceedings of the 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Abu Dhabi, United Arab Emirates, 14–18 October 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 6367–6374.
10. Li, H.; Zhang, H.; Zhang, Y.; Zhang, S.; Peng, Y.; Wang, Z.; Song, H.; Chen, M. An accurate activate screw detection method for automatic electric vehicle battery disassembly. *Batteries* **2023**, *9*, 187. [CrossRef]
11. Priyono, A.; Ijomah, W.; Bititci, U.S. Disassembly for remanufacturing: A systematic literature review, new model development and future research needs. *J. Ind. Eng. Manag.* **2016**, *9*, 899–932. [CrossRef]
12. Zang, Y.; Wang, Y. Robotic disassembly of electric vehicle batteries: An overview. In Proceedings of the 2022 27th International Conference on Automation and Computing (ICAC), Bristol, UK, 1–3 September 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–6.
13. Diekmann, J.; Grützke, M.; Loellhoeffel, T.; Petermann, M.; Rothermel, S.; Winter, M.; Nowak, S.; Kwade, A. Potential dangers during the handling of lithium-ion batteries. In *Recycling of Lithium-Ion Batteries: The LithoRec Way*; Springer: Cham, Switzerland, 2018; pp. 39–51.
14. Villagrossi, E.; Delledonne, M.; Faroni, M.; Beschi, M.; Pedrocchi, N. Hiding task-oriented programming complexity: An industrial case study. *Int. J. Comput. Integr. Manuf.* **2023**, *36*, 1629–1648. [CrossRef]
15. Baratta, A.; Cimino, A.; Gnoni, M.G.; Longo, F. Human Robot Collaboration in Industry 4.0: A literature review. *Procedia Comput. Sci.* **2023**, *217*, 1887–1895. [CrossRef] [PubMed]
16. Wegener, K.; Andrew, S.; Raatz, A.; Dröder, K.; Herrmann, C. Disassembly of electric vehicle batteries using the example of the Audi Q5 hybrid system. *Procedia Cirp* **2014**, *23*, 155–160. [CrossRef]
17. Wegener, K.; Chen, W.H.; Dietrich, F.; Dröder, K.; Kara, S. Robot assisted disassembly for the recycling of electric vehicle batteries. *Procedia Cirp* **2015**, *29*, 716–721. [CrossRef]
18. Duan, L.; Li, J.; Bao, J.; Lv, J.; Zheng, H. A MR-Assisted and Scene Perception System for Human-Robot Collaborative Disassembly of Power Batteries. In Proceedings of the 2023 IEEE 19th International Conference on Automation Science and Engineering (CASE), Auckland, New Zealand, 26–30 August 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1–8.
19. Wu, B.; Han, S.; Shin, K.G.; Lu, W. Application of artificial neural networks in design of lithium-ion batteries. *J. Power Sources* **2018**, *395*, 128–136. [CrossRef]
20. Meng, K.; Xu, G.; Peng, X.; Youcef-Toumi, K.; Li, J. Intelligent disassembly of electric-vehicle batteries: A forward-looking overview. *Resour. Conserv. Recycl.* **2022**, *182*, 106207. [CrossRef]
21. Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. Gpt-4 technical report. *arXiv* **2023**, arXiv:2303.08774. [CrossRef]
22. Anil, R.; Dai, A.M.; Firat, O.; Johnson, M.; Lepikhin, D.; Passos, A.; Shakeri, S.; Taropa, E.; Bailey, P.; Chen, Z.; et al. Palm 2 technical report. *arXiv* **2023**, arXiv:2305.10403. [CrossRef]
23. Kassianik, P.; Saglam, B.; Chen, A.; Nelson, B.; Vellore, A.; Aufiero, M.; Burch, F.; Kedia, D.; Zohary, A.; Weerawardhena, S.; et al. Llama-3.1-foundationai-securityllm-base-8b technical report. *arXiv* **2025**, arXiv:2504.21039.
24. Zeng, F.; Gan, W.; Wang, Y.; Liu, N.; Yu, P.S. Large language models for robotics: A survey. *arXiv* **2023**, arXiv:2311.07226. [CrossRef]
25. Wang, J.; Shi, E.; Hu, H.; Ma, C.; Liu, Y.; Wang, X.; Yao, Y.; Liu, X.; Ge, B.; Zhang, S. Large language models for robotics: Opportunities, challenges, and perspectives. *J. Autom. Intell.* **2024**, *4*, 52–64. [CrossRef]
26. Ahn, M.; Brohan, A.; Brown, N.; Chebotar, Y.; Cortes, O.; David, B.; Finn, C.; Fu, C.; Gopalakrishnan, K.; Hausman, K.; et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv* **2022**, arXiv:2204.01691. [CrossRef]
27. Driess, D.; Xia, F.; Sajjadi, M.S.; Lynch, C.; Chowdhery, A.; Wahid, A.; Tompson, J.; Vuong, Q.; Yu, T.; Huang, W.; et al. Palm-e: An embodied multimodal language model. *arXiv* **2023**, arXiv:2303.03378.
28. Singh, I.; Blukis, V.; Mousavian, A.; Goyal, A.; Xu, D.; Tremblay, J.; Fox, D.; Thomason, J.; Garg, A. Progprompt: Generating situated robot task plans using large language models. In Proceedings of the 2023 IEEE International Conference on Robotics and Automation (ICRA), London, UK, 29 May–2 June 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 11523–11530.
29. Singh, I.; Blukis, V.; Mousavian, A.; Goyal, A.; Xu, D.; Tremblay, J.; Fox, D.; Thomason, J.; Garg, A. ProgPrompt: Program generation for situated robot task planning using large language models. *Auton. Robot.* **2023**, *47*, 999–1012. [CrossRef]
30. Silver, T.; Hariprasad, V.; Shuttleworth, R.S.; Kumar, N.; Lozano-Pérez, T.; Kaelbling, L.P. PDDL planning with pretrained large language models. In *NeurIPS 2022 Foundation Models for Decision Making Workshop*; NeurIPS: San Diego, CA, USA, 2022.
31. Guan, L.; Valmeekam, K.; Sreedharan, S.; Kambhampati, S. Leveraging pre-trained large language models to construct and utilize world models for model-based task planning. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 79081–79094.
32. Kannan, S.S.; Venkatesh, V.L.; Min, B.C. Smart-llm: Smart multi-agent robot task planning using large language models. In Proceedings of the 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Abu Dhabi, United Arab Emirates, 14–18 October 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 12140–12147.

33. Fikes, R.E.; Nilsson, N.J. STRIPS: A new approach to the application of theorem proving to problem solving. *Artif. Intell.* **1971**, *2*, 189–208. [CrossRef]
34. Silver, T.; Athalye, A.; Tenenbaum, J.B.; Lozano-Pérez, T.; Kaelbling, L.P. Learning neuro-symbolic skills for bilevel planning. *arXiv* **2022**, arXiv:2206.10680. [CrossRef]
35. Martin, A.E.; Doumas, L.A. Predicate learning in neural systems: Using oscillations to discover latent structure. *Curr. Opin. Behav. Sci.* **2019**, *29*, 77–83. [CrossRef]
36. Doumas, L.A.; Puebla, G.; Martin, A.E. Human-like generalization in a machine through predicate learning. *arXiv* **2018**, arXiv:1806.01709.
37. Han, M.; Zhu, Y.; Zhu, S.C.; Wu, Y.N.; Zhu, Y. Interpret: Interactive predicate learning from language feedback for generalizable task planning. *arXiv* **2024**, arXiv:2405.19758.
38. Hussein, A.; Gaber, M.M.; Elyan, E.; Jayne, C. Imitation learning: A survey of learning methods. *ACM Comput. Surv.* **2017**, *50*, 1–35. [CrossRef]
39. Schaal, S. Is imitation learning the route to humanoid robots? *Trends Cogn. Sci.* **1999**, *3*, 233–242. [CrossRef]
40. Duan, Y.; Andrychowicz, M.; Stadie, B.; Jonathan Ho, O.; Schneider, J.; Sutskever, I.; Abbeel, P.; Zaremba, W. One-shot imitation learning. *arXiv* **2017**, arXiv:1703.07326. [CrossRef]
41. Jansonnie, P.; Wu, B.; Perez, J.; Peters, J. Unsupervised Skill Discovery for Robotic Manipulation through Automatic Task Generation. In Proceedings of the 2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids), Nancy, France, 22–24 November 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 926–933.
42. Lynch, C.; Khansari, M.; Xiao, T.; Kumar, V.; Tompson, J.; Levine, S.; Sermanet, P. Learning latent plans from play. In Proceedings of the Conference on Robot Learning, virtual, 16–18 November 2020; pp. 1113–1132.
43. Jin, W.; Cheng, Y.; Shen, Y.; Chen, W.; Ren, X. A good prompt is worth millions of parameters: Low-resource prompt-based learning for vision-language models. *arXiv* **2021**, arXiv:2110.08484.
44. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D.; Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 24824–24837.
45. He, J.; Rungta, M.; Koleczek, D.; Sekhon, A.; Wang, F.X.; Hasan, S. Does Prompt Formatting Have Any Impact on LLM Performance? *arXiv* **2024**, arXiv:2411.10541. [CrossRef]
46. Qian, S.; Ning, C.; Hu, Y. MobileNetV3 for image classification. In Proceedings of the 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Nanchang, China, 26–28 March 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 490–497.
47. Koonce, B. MobileNetV3. In *Convolutional nEural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 125–144.
48. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
49. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 116–131.
50. Xu, H.; Ghosh, G.; Huang, P.Y.; Arora, P.; Aminzadeh, M.; Feichtenhofer, C.; Metze, F.; Zettlemoyer, L. Vlm: Task-agnostic video-language model pre-training for video understanding. *arXiv* **2021**, arXiv:2105.09996.