Large Language Models in Robotics: A Comprehensive Literature Review

Combining Curative Paper Lists with Simple Explanations

Research Documentation

December 2025

Abstract

This document presents a comprehensive review of how Large Language Models (LLMs) are being used to make robots smarter. LLMs are AI systems (like ChatGPT) that understand and generate human language. Researchers are now connecting these language models to robots so that robots can understand spoken commands, plan tasks, navigate spaces, and interact with humans more naturally. This review covers 80+ important research papers across seven main areas: (1) Surveys, (2) Reasoning, (3) Planning, (4) Manipulation, (5) Navigation and Instructions, (6) Simulation Frameworks, and (7) Safety and Adversarial Testing.

Contents

1 Introduction: Why Use LLMs in Robots?

Traditional robots need to be programmed with exact instructions for every situation. This is hard because the real world is unpredictable. By connecting an LLM to a robot, the robot can:

- Understand commands in natural language (like "bring me something to drink")
- Use common sense knowledge (knowing that a fridge usually contains drinks)
- Break down big tasks into smaller steps
- Explain what it's doing and why

The Basic Architecture: Most LLM-robot systems work like this:

1. **Human speaks**: "Clean up the kitchen table"
2. **LLM thinks**: Breaks this into steps: find objects, pick them up, put them away
3. **Robot acts**: Executes each step using its cameras and arms
4. **Feedback loop**: If something goes wrong, the LLM replans

2 Surveys and Overview Papers

Survey papers provide comprehensive overviews of the field and are essential reading for understanding the landscape.

- **A Survey of Robotic Language Grounding**: "Tradeoffs between Symbols and Embeddings" – *IJCAI, Aug 2024*. arXiv:2405.13245
- **A Superalignment Framework in Autonomous Driving**: Using Large Language Models – *arXiv, Jun 2024*. arXiv:2406.05651
- **Neural Scaling Laws for Embodied AI** – *arXiv, May 2024*. arXiv:2405.14005
- **On the Prospects of LLMs in Automated Planning and Scheduling** – *ICAPS, May 2024*
- **Toward General-Purpose Robots via Foundation Models**: A Survey and Meta-Analysis – *arXiv, Dec 2023*. arXiv:2312.08782
- **Language-conditioned Learning for Robotic Manipulation**: A Survey – *arXiv, Dec 2023*. arXiv:2312.10807
- **Foundation Models in Robotics**: Applications, Challenges, and the Future – *arXiv, Dec 2023*. arXiv:2312.07843
- **Robot Learning in the Era of Foundation Models**: A Survey – *arXiv, Nov 2023*. arXiv:2311.14379
- **The Development of LLMs for Embodied Navigation** – *arXiv, Nov 2023*. arXiv:2311.00530

3 Reasoning

Reasoning involves understanding context, making inferences, and connecting language understanding to physical actions.

3.1 Recent Advances (2024-2025)

- **RoboRefer** (June 2025): "Towards Spatial Referring with Reasoning in Vision-Language Models for Robotics" – arXiv:2506.04308
- **RoboSpatial** (CVPR, June 2025): "Teaching Spatial Understanding to 2D and 3D Vision-Language Models for Robotics" – arXiv:2411.16537
- **RobotxR1** (May 2025): "Enabling Embodied Robotic Intelligence on LLMs through Closed-Loop Reinforcement Learning" – arXiv:2505.03238
- **SPINE** (ICRA, May 2025): "Online Semantic Planning for Missions with Incomplete Natural Language Specifications" – arXiv:2410.03035

- **ELLMER** (Nature Machine Intelligence, Mar 2025): "Embodied large language models enable robots to complete long-horizon tasks in unpredictable settings"
- **AHA** (Oct 2024): "A Vision-Language-Model for Detecting and Reasoning over Failures in Robotic Manipulation" – arXiv:2410.00371
- **ReKep** (Sep 2024): "Spatio-Temporal Reasoning of Relational Keypoint Constraints for Robotic Manipulation" – arXiv:2409.01652
- **Octopi** (RSS, June 2024): "Object Property Reasoning with Large Tactile-Language Models" – arXiv:2405.02794
- **AutoRT** (Jan 2024): "Embodied Foundation Models for Large Scale Orchestration of Robotic Agents" – arXiv:2401.12963

3.2 Foundational Reasoning Papers (2022-2023)

3.2.1 SayCan (Google, 2022)

Paper: "Do As I Can, Not As I Say: Grounding Language in Robotic Affordances" – arXiv:2204.01691

Simple Explanation: Imagine you ask your robot "I'm hungry." The robot needs to figure out what it CAN actually do to help. SayCan combines what the LLM SUGGESTS (like "bring food from the kitchen") with what the robot can ACTUALLY DO (its "affordances"). If the robot can't reach the top shelf, SayCan won't suggest getting food from there, even if the LLM thinks that's where the food is. It's like having a smart friend who gives advice but checks first if you can actually follow it.

Why It's Important: This was the first major paper showing that LLMs can work with real robots by checking what's physically possible.

3.2.2 Socratic Models (Google, 2022)

Paper: "Composing Zero-Shot Multimodal Reasoning with Language" – arXiv:2204.00598

Simple Explanation: Named after Socrates who taught through questions, this system has multiple AI models "discuss" problems. A vision model sees an image and describes it. A language model reasons about the description. An audio model processes speech. They pass information back and forth like experts in a meeting. No single model does everything - they combine their strengths through dialogue.

Why It's Important: Shows how different AI specialists can work together through language.

3.2.3 PaLM-E (Google, 2023)

Paper: "PaLM-E: An Embodied Multimodal Language Model" – arXiv:2303.03378

Simple Explanation: PaLM-E is huge - 562 billion parameters (ChatGPT has about 175 billion). It can see images, understand language, and control robots all in one model. Feed it a picture of a kitchen and ask "what can I make for breakfast?" It sees the eggs, pan, and stove and suggests omelets with a plan to make them. It directly understands the visual world and connects it to robot actions.

Why It's Important: Single massive model that handles vision, language, and robot control together.

4 Planning

Task planning means figuring out HOW to do something. When you ask a robot to "make breakfast," it needs to plan: first get eggs, then heat the pan, then crack eggs, etc. LLMs are very good at this because they know about everyday activities from the text they learned from.

4.1 Recent Planning Advances (2024-2025)

- **FLARE** (AAAI, Mar 2025): "Multi-Modal Grounded Planning and Efficient Replanning For Learning Embodied Agents" – arXiv:2412.17288
- **LLM+MAP** (Mar 2025): "Bimanual Robot Task Planning using Large Language Models and PDDL" – arXiv:2503.17309
- **Code-as-Monitor** (CVPR, 2025): "Constraint-aware Visual Programming for Reactive and Proactive Robotic Failure Detection" – arXiv:2412.04455

- **LABOR Agent** (Humanoids, Nov 2024): "Large Language Models for Orchestrating Bimanual Robots" – arXiv:2404.02018
- **SELP** (Sep 2024): "Generating Safe and Efficient Task Plans for Robot Agents with LLMs" – arXiv:2409.19471
- **Wonderful Team** (Jul 2024): "Solving Robotics Problems in Zero-Shot with Vision-Language Models" – arXiv:2407.19094
- **FLTRNN** (ICRA, May 2024): "Faithful Long-Horizon Task Planning for Robotics with Large Language Models"
- **LLM-Personalize** (Apr 2024): "Aligning LLM Planners with Human Preferences via Reinforced Self-Training" – arXiv:2404.14285
- **LLM3** (IROS, Mar 2024): "Large Language Model-based Task and Motion Planning with Motion Failure Reasoning" – arXiv:2403.11552
- **SayCanPay** (AAAI, Jan 2024): "Heuristic Planning with Large Language Models Using Learnable Domain Knowledge" – arXiv:2308.12682

4.2 Core Planning Papers with Explanations

4.2.1 Code-as-Policies (Google, 2022)

Paper: "Code as Policies: Language Model Programs for Embodied Control" – arXiv:2209.07753

Simple Explanation: Instead of the LLM just saying what to do in words, this paper has the LLM write actual computer code (Python) that controls the robot. So when you say "move in a circle," the LLM writes code like: "for angle in range(360): move_forward(0.1); turn(1)". This is more precise than just giving instructions in English.

Why It's Important: Writing code is more precise and can handle complex math and logic that simple language instructions can't.

4.2.2 Inner Monologue (Google, 2022)

Paper: "Embodied Reasoning through Planning with Language Models" – arXiv:2207.05608

Simple Explanation: This paper gives the robot an "inner voice" - it talks to itself! When the robot tries to pick up a cup but fails, it gets feedback: "gripper empty, cup still on table." The LLM then reasons: "I missed the cup, let me try again but position my arm differently."

Why It's Important: Robots make mistakes. This paper shows how LLMs can use feedback to recover from errors.

4.2.3 ReAct (Princeton/Google, 2022)

Paper: "Synergizing Reasoning and Acting in Language Models" – arXiv:2210.03629

Simple Explanation: ReAct makes the LLM alternate between THINKING and ACTING. Think: "I need to find the remote." Act: Look around the room. Think: "I see the sofa, remotes are often near sofas." Act: Go to the sofa. This back-and-forth helps the robot stay on track because it keeps checking its reasoning against what it sees in the real world.

Why It's Important: This prevents the robot from making a wrong plan and blindly following it.

4.2.4 LLM-Planner (Ohio State, 2022)

Paper: "Few-Shot Grounded Planning for Embodied Agents with Large Language Models" – arXiv:2212.04088

Simple Explanation: Most robot systems need thousands of examples to learn. LLM-Planner needs very few (that's what "few-shot" means). You show it just 3-5 examples of tasks, and it can plan new tasks it has never seen. It also updates its plan when things change.

Why It's Important: Robots need to work in new environments without tons of training examples.

4.2.5 ProgPrompt (NVIDIA/Cornell, 2022)

Paper: "Generating Situated Robot Task Plans using Large Language Models" – arXiv:2209.11302

Simple Explanation: This paper creates a special way to talk to the LLM using programming-style prompts. You tell it what actions the robot can do (like grab(), move(), place()) and what objects exist in the room. Then the LLM generates a plan using only these available actions.

Why It's Important: Makes sure the LLM only suggests actions the robot can actually perform.

4.2.6 VOYAGER (NVIDIA/Caltech, 2023)

Paper: "An Open-Ended Embodied Agent with Large Language Models" – arXiv:2305.16291

Simple Explanation: VOYAGER is a Minecraft AI that learns forever (that's "lifelong learning"). It plays the game, learns new skills, saves them in a "skill library," and builds on them. First it learns to chop wood, then craft tools, then build houses. Each new skill uses the old ones.

Why It's Important: Shows how robots can keep learning and improving without limits.

4.2.7 Language Models as Zero-Shot Planners (Berkeley, 2022)

Paper: "Extracting Actionable Knowledge for Embodied Agents" – arXiv:2201.07207

Simple Explanation: "Zero-shot" means the robot plans without ANY examples for that specific task. If you say "make coffee," it plans the steps even if it has never been taught about making coffee specifically. The LLM already knows about coffee-making from all the text it read.

Why It's Important: This foundational paper showed that LLMs already know how to do many tasks - we just need to extract that knowledge.

4.2.8 SayPlan (Queensland, 2023)

Paper: "Grounding Large Language Models using 3D Scene Graphs for Scalable Task Planning" – arXiv:2307.06135

Simple Explanation: SayPlan works in HUGE buildings with many floors and rooms. It uses a "scene graph" - a map showing all objects and rooms and how they're connected. The LLM can search this map to find things. Works in offices, hospitals, big buildings.

Why It's Important: Makes LLM planning work in large, complex environments, not just small rooms.

4.3 Additional Planning Papers

- **ViLa** (Sep 2023): "Unveiling the Power of GPT-4V in Robotic Vision-Language Planning" – arXiv:2311.17842
- **CoPAL** (ICRA, Oct 2023): "Corrective Planning of Robot Actions with LLMs" – arXiv:2310.07263
- **LGMCTS** (Sep 2023): "Language-Guided Monte-Carlo Tree Search for Executable Semantic Object Rearrangement" – arXiv:2309.15821
- **Prompt2Walk** (Sep 2023): "Prompt a Robot to Walk with Large Language Models" – arXiv:2309.09969
- **DoReMi** (July 2023): "Grounding Language Model by Detecting and Recovering from Plan-Execution Misalignment" – arXiv:2307.00329
- **Co-LLM-Agents** (Jul 2023): "Building Cooperative Embodied Agents Modularly with LLMs" – arXiv:2307.02485
- **LLM-Reward** (Jun 2023): "Language to Rewards for Robotic Skill Synthesis" – arXiv:2306.08647
- **LLM-BRAIn** (May 2023): "AI-driven Fast Generation of Robot Behaviour Tree based on LLM" – arXiv:2305.19352
- **LLM-MCTS** (May 2023): "Large Language Models as Commonsense Knowledge for Large-Scale Task Planning" – arXiv:2305.14078
- **LLM+P** (Apr 2023): "Empowering Large Language Models with Optimal Planning Proficiency" – arXiv:2304.11477
- **ChatGPT for Robotics** (Microsoft, Feb 2023): "Design Principles and Model Abilities" – arXiv:2306.17582
- **Gato** (DeepMind, Nov 2022): "A Generalist Agent" – arXiv:2205.06175

5 Manipulation

Manipulation means using robot arms and grippers to interact with objects - picking things up, moving them, using tools. LLMs help by understanding WHAT to do with objects based on their meaning, not just their shape.

5.1 Recent Manipulation Papers (2024-2025)

- **LLM-TALE** (Sep 2025): "LLM-Guided Task- and Affordance-Level Exploration in RL" – arXiv:2509.16615
- **Meta-Control** (CoRL, Nov 2024): "Automatic Model-based Control System Synthesis for Heterogeneous Robot Skills" – arXiv:2405.11380
- **A3VLM** (CoRL, Nov 2024): "Actionable Articulation-Aware Vision Language Model" – arXiv:2406.07549
- **Manipulate-Anything** (CoRL, Nov 2024): "Automating Real-World Robots using Vision-Language Models" – arXiv:2406.18915
- **RobiButler** (Sep 2024): "Remote Multimodal Interactions with Household Robot Assistant" – arXiv:2409.20548
- **SKT** (Sep 2024): "Integrating State-Aware Keypoint Trajectories with VLMs for Robotic Garment Manipulation" – arXiv:2409.18082
- **Plan-Seq-Learn** (ICLR, May 2024): "Language Model Guided RL for Solving Long Horizon Robotics Tasks" – arXiv:2405.01534
- **ExploRLLM** (Mar 2024): "Guiding Exploration in Reinforcement Learning with Large Language Models" – arXiv:2403.09583
- **ManipVQA** (IROS, Mar 2024): "Injecting Robotic Affordance and Physically Grounded Information into Multi-Modal LLMs" – arXiv:2403.11289

5.2 Core Manipulation Papers with Explanations

5.2.1 VoxPoser (Stanford, 2023)

Paper: "Composable 3D Value Maps for Robotic Manipulation with Language Models" – arXiv:2307.05973

Simple Explanation: VoxPoser creates 3D "value maps" that tell the robot where to go and where to avoid. If you say "open the drawer," the LLM thinks: "The handle has high value (go there), the drawer body has low value (avoid hitting it)." These values are painted onto a 3D grid (voxels), and the robot follows the high-value path.

Why It's Important: Translates language instructions into spatial guidance that robot arms can follow.

5.2.2 TidyBot (Princeton, 2023)

Paper: "Personalized Robot Assistance with Large Language Models" – arXiv:2305.05658

Simple Explanation: TidyBot learns YOUR preferences for tidying up. Does the TV remote belong on the couch or in a drawer? TidyBot asks a few questions, remembers your answers, and then tidies according to YOUR rules. The LLM helps it generalize: if you like books on shelves, it guesses you might like magazines there too.

Why It's Important: Personalization - the robot adapts to individual users rather than following fixed rules.

5.2.3 CLIPort (Washington/NVIDIA, 2021)

Paper: "What and Where Pathways for Robotic Manipulation" – arXiv:2109.12098

Simple Explanation: CLIPort answers two questions: WHAT to interact with and WHERE exactly to grab. It uses CLIP (a vision-language model) to understand "what" - finding the right object from a description. It uses Transporter networks to understand "where" - the exact pixel location to pick and place.

Why It's Important: Combines semantic understanding (what) with spatial precision (where) for accurate manipulation.

5.2.4 GraspGPT (PKU, 2023)

Paper: "Leveraging Semantic Knowledge from a Large Language Model for Task-Oriented Grasping" – arXiv:2307.13204

Simple Explanation: Different tasks require grabbing objects differently. To pour water, grab the cup by the handle. To hand it to someone, maybe grab the body. GraspGPT uses LLM knowledge to choose the right grasp for the task. If you say "use the hammer to hit a nail," it knows to grab the handle, not the head.

Why It's Important: Task-appropriate grasping - understanding that HOW you grab depends on WHAT you're trying to do.

5.3 Additional Manipulation Papers

- **BOSS** (CoRL, Nov 2023): "Bootstrap Your Own Skills: Learning to Solve New Tasks with LLM Guidance"
- **Lafite-RL** (CoRL Workshop, Nov 2023): "Accelerating RL of Robotic Manipulations via Feedback from LLMs" – arXiv:2311.02379
- **Octopus** (Oct 2023): "Embodied Vision-Language Programmer from Environmental Feedback" – arXiv:2310.08588
- **Text2Reward** (Sep 2023): "Automated Dense Reward Function Generation for RL" – arXiv:2309.11489
- **PhysObjects** (Sep 2023): "Physically Grounded Vision-Language Models for Robotic Manipulation" – arXiv:2309.02561
- **Scalingup** (July 2023): "Language-Guided Robot Skill Acquisition" – arXiv:2307.14535
- **LIV** (Jun 2023): "Language-Image Representations and Rewards for Robotic Control" – arXiv:2306.00958
- **RoboCat** (DeepMind, Jun 2023): "A self-improving robotic agent" – arXiv:2306.11706
- **Grasp Anything** (June 2023): "Transferring Foundation Models for Universal Pick-Place Robots" – arXiv:2306.05716
- **Instruct2Act** (May 2023): "Mapping Multi-modality Instructions to Robotic Actions with LLM" – arXiv:2305.11176
- **VIMA** (Oct 2022): "General Robot Manipulation with Multimodal Prompts" – arXiv:2210.03094
- **Perceiver-Actor** (CoRL, Sep 2022): "A Multi-Task Transformer for Robotic Manipulation" – arXiv:2209.05451
- **R3M** (Nov 2022): "A Universal Visual Representation for Robot Manipulation" – arXiv:2203.12601

6 Instructions and Navigation

Navigation is about getting from point A to point B safely. LLMs help robots understand directions like "go past the blue door and turn left at the plant" - things that are easy for humans but hard for traditional robots.

6.1 Recent Navigation Papers (2024-2025)

- **LLMxRobot** (RSS, Apr 2025): "Autonomous Driving Systems with On-Board LLMs"
- **GSON** (Sep 2024): "A Group-based Social Navigation Framework with Large Multimodal Model" – arXiv:2409.18084
- **Navid** (Mar 2024): "Video-based VLM Plans the Next Step for Vision-and-Language Navigation" – arXiv:2402.15852
- **OVSG** (CoRL, Nov 2023): "Context-Aware Entity Grounding with Open-Vocabulary 3D Scene Graphs"

6.2 Core Navigation Papers with Explanations

6.2.1 LM-Nav (Berkeley, 2022)

Paper: "Robotic Navigation with Large Pre-trained Models of Language, Vision, and Action" – arXiv:2207.04429

Simple Explanation: LM-Nav combines three AI systems: one that understands language (GPT-3), one that understands images (CLIP), and one that controls movement. You say "go to the big rock near the tree." GPT-3 understands this means finding something stone-like near vegetation. CLIP looks at camera images and finds things matching that description.

Why It's Important: First system combining multiple AI models for outdoor navigation with language commands.

6.2.2 VLMaps (Freiburg, 2022)

Paper: "Visual Language Maps for Robot Navigation" – arXiv:2210.05714

Simple Explanation: VLMaps creates a special kind of map. Normal robot maps just show walls and obstacles. VLMaps adds meaning: "this area is the kitchen," "here's a sofa," "that's a window." Now you can say "go between the sofa and TV" and the robot understands because it has a map with these labels.

Why It's Important: Robots can understand location commands in natural language, not just coordinates.

6.2.3 NavGPT (Adelaide, 2023)

Paper: "Explicit Reasoning in Vision-and-Language Navigation with Large Language Models" – arXiv:2305.16986

Simple Explanation: NavGPT makes the robot's navigation reasoning visible. At each step, it explains: "I see a hallway and a door. The instruction says go to the bedroom. Bedrooms are usually through doors, not in hallways. I'll go through the door." You can see exactly WHY the robot made each choice.

Why It's Important: Shows the reasoning process, making navigation decisions transparent and debuggable.

6.2.4 CLIP-Fields (NYU, 2022)

Paper: "Weakly Supervised Semantic Fields for Robotic Memory" – arXiv:2210.05663

Simple Explanation: CLIP-Fields creates a robot's memory of a place. As the robot explores, it remembers what it saw and where. Later, you can ask "where did you see a red mug?" and it recalls the location. It's like giving the robot a photographic memory that you can search with questions.

Why It's Important: Robots can build searchable memories of environments without human supervision.

6.3 Additional Navigation Papers

- **Interactive Language** (Oct 2022): "Talking to Robots in Real Time" – arXiv:2210.06407
- **NLMap** (Sep 2022): "Open-vocabulary Queryable Scene Representations for Real World Planning" – arXiv:2209.09874
- **ADAPT** (CVPR, May 2022): "Vision-Language Navigation with Modality-Aligned Action Prompts" – arXiv:2205.15509
- **CoW** (Mar 2022): "CLIP on Wheels: Zero-Shot Object Navigation as Object Localization and Exploration" – arXiv:2203.10421
- **Recurrent VLN-BERT** (CVPR, Jun 2021): "A Recurrent Vision-and-Language BERT for Navigation" – arXiv:2011.13922
- **VLN-BERT** (ECCV, Apr 2020): "Improving Vision-and-Language Navigation with Image-Text Pairs from the Web" – arXiv:2004.14973

7 Scene Understanding and Perception

Before acting, robots must understand what they see. LLMs help interpret scenes, identify objects, understand relationships, and reason about the physical world.

7.1 Vision-Language-Action Models

7.1.1 RT-2 (Google DeepMind, 2023)

Paper: "Vision-Language-Action Models Transfer Web Knowledge to Robotic Control" – arXiv:2307.15818

Simple Explanation: RT-2 is a Vision-Language-Action (VLA) model. It learned from internet images and text, then learned to control robots. The amazing thing: knowledge transfers! Because it saw pictures of lions online, it can identify a toy lion it has never seen and "put it with the other animals." Web knowledge helps with robot tasks.

Why It's Important: Shows that web-scale training helps robots handle novel situations.

7.1.2 RT-1 (Google, 2022)

Paper: "Robotics Transformer for Real-World Control at Scale" – arXiv:2212.06817

Simple Explanation: RT-1 is a Transformer (like ChatGPT) trained on 130,000 robot demonstrations doing 700+ tasks. It sees camera images, reads a task instruction, and outputs motor commands. Because it was trained on so many examples, it generalizes well to new tasks and objects. It works at 3 actions per second - fast enough for real-time control.

Why It's Important: First large-scale robot transformer trained on real robot data, not simulations.

7.1.3 Open X-Embodiment (21 Institutions, 2023)

Paper: "Robotic Learning Datasets and RT-X Models" – arXiv:2310.08864

Simple Explanation: This is a massive collaboration: 21 research labs shared their robot data - over 1 million robot experiences from 22 different robot types. They trained RT-X models on all this data. The cool finding: a model trained on many different robots actually works BETTER on each individual robot than models trained on just that robot's data.

Why It's Important: Largest robot learning dataset ever; proves cross-robot knowledge transfer works.

7.1.4 ConceptGraphs (MIT/Toronto, 2023)

Paper: "Open-Vocabulary 3D Scene Graphs for Perception and Planning" – arXiv:2309.16650

Simple Explanation: ConceptGraphs builds a 3D map where objects are nodes and relationships are edges. "Cup is ON table. Table is NEAR chair. Chair is IN kitchen." You can ask questions like "what's near the window?" or "find something to drink" and it searches this graph.

Why It's Important: Creates queryable, semantic 3D maps for robot reasoning and planning.

7.2 Additional Perception Papers

- **LEO** (Nov 2023): "An Embodied Generalist Agent in 3D World" – arXiv:2311.12871
- **LLaRP** (Oct 2023): "Large Language Models as Generalizable Policies for Embodied Tasks" – arXiv:2310.17722
- **CortexBench** (Mar 2023): "Where are we in the search for an Artificial Visual Cortex for Embodied Intelligence?" – arXiv:2303.18240
- **Matcha** (IROS, Mar 2023): "Chat with the Environment: Interactive Multimodal Perception using LLMs" – arXiv:2303.08268
- **Generative Agents** (Apr 2023): "Interactive Simulacra of Human Behavior" – arXiv:2304.03442
- **MetaMorph** (Mar 2022): "Learning Universal Controllers with Transformers" – arXiv:2203.11931
- **Embodied-CLIP** (CVPR, Nov 2021): "Simple but Effective: CLIP Embeddings for Embodied AI" – arXiv:2111.09888

8 Simulation Frameworks

Simulation environments are essential for training and testing LLM-powered robots before deployment in the real world.

- **ManiSkill3** (Oct 2024): "GPU Parallelized Robotics Simulation and Rendering for Generalizable Embodied AI" – arXiv:2410.00425
- **GENESIS** (Nov 2023): "A generative world for general-purpose robotics & embodied AI learning"
- **ARNOLD** (ICCV, Apr 2023): "A Benchmark for Language-Grounded Task Learning With Continuous States in Realistic 3D Scenes" – arXiv:2304.04321
- **OmniGibson** (CoRL, 2022): "A platform for accelerating Embodied AI research built upon NVIDIA's Omniverse engine"
- **MineDojo** (Jun 2022): "Building Open-Ended Embodied Agents with Internet-Scale Knowledge" – arXiv:2206.08853
- **Habitat 2.0** (NeurIPS, Dec 2021): "Training Home Assistants to Rearrange their Habitat" – arXiv:2106.14405
- **BEHAVIOR** (CoRL, Nov 2021): "Benchmark for Everyday Household Activities in Virtual, Interactive, and Ecological Environments" – arXiv:2108.03332
- **iGibson 1.0** (IROS, Sep 2021): "A Simulation Environment for Interactive Tasks in Large Realistic Scenes" – arXiv:2012.02924
- **ALFRED** (CVPR, Jun 2020): "A Benchmark for Interpreting Grounded Instructions for Everyday Tasks" – arXiv:1912.01734
- **BabyAI** (ICLR, May 2019): "A Platform to Study the Sample Efficiency of Grounded Language Learning" – arXiv:1810.08272

9 Safety, Risks, and Adversarial Testing

Safety is critical. Robots must avoid hurting people, breaking things, and getting stuck. This section covers papers on safety guarantees, adversarial testing, and risk mitigation.

9.1 Safety and Adversarial Papers

- **BadVLA** (May 2025): "Towards Backdoor Attacks on Vision-Language-Action Models via Objective-Decoupled Optimization" – arXiv:2505.16640
- **RoboPAIR** (ICRA, May 2025): "Jailbreaking LLM-Controlled Robots" – arXiv:2410.13691
- **RoboGuard** (Apr 2025): "Safety Guardrails for LLM-Enabled Robots" – arXiv:2503.07885
- **Safe LLM-Controlled Robots with Formal Guarantees** (Mar 2025): "via Reachability Analysis" – arXiv:2503.03911
- **LLM-Driven Robots Risk Enacting Discrimination** (Jun 2024): "Violence, and Unlawful Actions" – arXiv:2406.08824
- **Highlighting the Safety Concerns of Deploying LLMs/VLMs in Robotics** (Feb 2024) – arXiv:2402.10340
- **Exploring Adversarial Vulnerabilities of VLA Models** (Nov 2024) – arXiv:2411.13587
- **Robots Enact Malignant Stereotypes** (FAccT, Jun 2022) – arXiv:2207.11569

9.2 Evaluation and Benchmarking

9.2.1 OK-Robot (NYU/Meta, 2024)

Paper: "What Really Matters in Integrating Open-Knowledge Models for Robotics" – arXiv:2401.12202

Simple Explanation: OK-Robot is a practical system that works in REAL homes (not just labs). It combines vision models for finding objects, navigation for moving around, and grasping for picking things up. In 10 real homes, it achieved 58.5% success on pick-and-place tasks - and 82% in clean environments. The paper reveals what actually matters in practice: sensor calibration, careful positioning, and handling clutter.

Why It's Important: Honest evaluation in real messy homes, not perfect lab conditions.

10 Key Challenges and Limitations

While LLMs make robots smarter, serious challenges remain:

1. **Hallucination**: LLMs sometimes make up facts. If an LLM imagines a door that doesn't exist, the robot will crash trying to go through it.
2. **Physical Understanding**: LLMs learned from text, not physics. They don't truly understand that heavy objects fall or that glass breaks.
3. **Latency**: LLMs are slow. If a ball is flying at the robot, it can't wait 2 seconds for the LLM to think. Fast reactions need different approaches.
4. **Social Intelligence**: Understanding when someone is uncomfortable, recognizing sarcasm, or knowing when to ask for help - these remain very difficult.
5. **Long-horizon Tasks**: Tasks with many steps over long times (like cooking a full meal) remain challenging because errors compound.
6. **Safety and Alignment**: Ensuring robots don't perform harmful actions, even when given adversarial prompts.

11 Complete Reference Table

The following table contains key papers organized by category with their arXiv IDs and publication years.

Paper Name	arXiv ID	Full Title	Year
Reasoning			
SayCan	2204.01691	Do As I Can, Not As I Say	2022
Socratic Models	2204.00598	Composing Zero-Shot Multimodal Reasoning	2022
PaLM-E	2303.03378	An Embodied Multimodal Language Model	2023
ReKep	2409.01652	Spatio-Temporal Reasoning of Relational Keypoint	2024
Planning			
Code-as-Policies	2209.07753	Language Model Programs for Embodied Control	2022
Inner Monologue	2207.05608	Embodied Reasoning through Planning	2022
ReAct	2210.03629	Synergizing Reasoning and Acting	2022
LLM-Planner	2212.04088	Few-Shot Grounded Planning	2022
ProgPrompt	2209.11302	Generating Situated Robot Task Plans	2022
VOYAGER	2305.16291	Open-Ended Embodied Agent	2023
Zero-Shot Planners	2201.07207	Extracting Actionable Knowledge	2022

Paper Name	arXiv ID	Full Title	Year
SayPlan	2307.06135	Grounding LLMs using 3D Scene Graphs	2023
ChatGPT for Robotics	2306.17582	Design Principles and Model Abilities	2023
Manipulation			
VoxPoser	2307.05973	Composable 3D Value Maps	2023
TidyBot	2305.05658	Personalized Robot Assistance	2023
CLIPort	2109.12098	What and Where Pathways	2021
GraspGPT	2307.13204	Task-Oriented Grasping	2023
VIMA	2210.03094	Robot Manipulation with Multi-modal Prompts	2022
Navigation			
LM-Nav	2207.04429	Navigation with Pre-trained Models	2022
VLMaps	2210.05714	Visual Language Maps	2022
NavGPT	2305.16986	Explicit Reasoning in VLN	2023
CLIP-Fields	2210.05663	Semantic Fields for Robotic Memory	2022
Scene Understanding			
RT-2	2307.15818	Vision-Language-Action Models	2023
RT-1	2212.06817	Robotics Transformer at Scale	2022
Open X-Embodiment	2310.08864	Robotic Learning Datasets and RT-X	2023
ConceptGraphs	2309.16650	Open-Vocabulary 3D Scene Graphs	2023
Safety and Evaluation			
OK-Robot	2401.12202	Integrating Open-Knowledge Models	2024
RoboPAIR	2410.13691	Jailbreaking LLM-Controlled Robots	2025
RoboGuard	2503.07885	Safety Guardrails for LLM Robots	2025