Toward General-Purpose Robots via Foundation Models: A Survey and Meta-Analysis

Yafei Hu[1]* Quanting Xie[1]* Vidhi Jain[1]*

Jonathan Francis[1,2] Jay Patrikar[1] Nikhil Keetha[1] Seungchan Kim[1] Yaqi Xie[1] Tianyi Zhang[1]

Shibo Zhao[1] Yu Quan Chong[1] Chen Wang[3] Katia Sycara[1] Matthew Johnson-Roberson[1]

Dhruv Batra[4,5] Xiaolong Wang[6] Sebastian Scherer[1] Zsolt Kira[4] Fei Xia[7†] Yonatan Bisk[1,5†]

[1]CMU [2]Bosch Center for AI [3]SAIR Lab [4]Georgia Tech [5]FAIR at Meta [6]UC San Diego [7]Google DeepMind

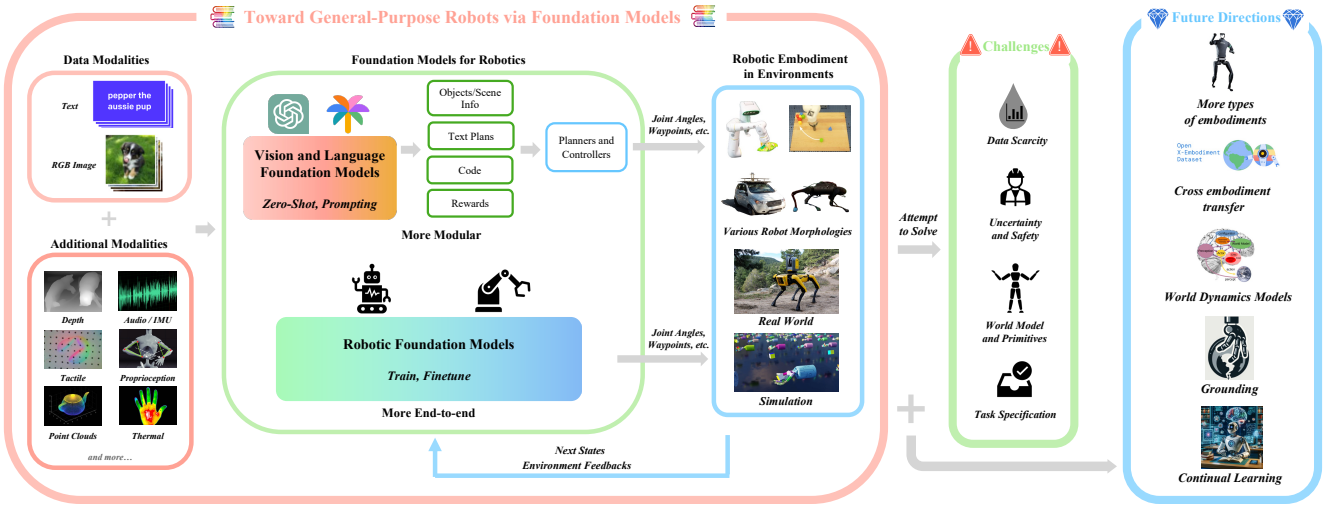


Figure 1: In this paper, we present a survey toward *building general-purpose robots via foundation models*. We mainly categorize the foundation models into vision and language models used in robotics, and robotic foundation models. We also introduce how these models could mitigate the challenges of classical robotic challenges, and projections of the potential future research directions. [1]

Abstract

Building general-purpose robots that can operate seamlessly, in any environment, with any object, and utilizing various skills to complete diverse tasks has been a long-standing goal in Artificial Intelligence. Unfortunately, however, most existing robotic systems have been constrained—having been designed for specific tasks, trained on specific datasets, and deployed within specific environments. These systems usually require extensively-labeled data, rely on task-specific models, have numerous generalization issues when deployed in real-world scenarios, and struggle to remain robust to distribution shifts. Motivated by the impressive open-set performance and content generation capabilities of web-scale, large-capacity pre-trained models (i.e., **foundation models**) in research fields such as Natural Language Processing (NLP) and Computer Vision (CV), we devote this survey to exploring (i) how these existing foundation models from NLP and CV can be applied to the field of robotics, and also exploring (ii) what a robotics-specific foundation model would look like. We begin by providing an overview of what constitutes a conventional robotic system and the fundamental barriers to making it universally applicable. Next, we establish a taxonomy to discuss current work exploring ways to leverage existing foundation models for robotics and develop ones catered to robotics. Finally, we discuss key challenges and promising future directions in using foundation models for enabling **general-purpose robotic systems**. We encourage readers to view our living GitHub repository[2] of resources, including papers reviewed in this survey as well as related projects and repositories for developing foundation models for robotics: **https://robotics-fm-survey.github.io/**.

*Equal contribution. {yafeih, quantinx, vidhij}@andrew.cmu.edu

†Equal advising. xiafei@google.com, ybisk@cs.cmu.edu

[1]Some images in this paper are screenshots from the papers we surveyed, icon images from Microsoft PowerPoint and MacOS Keynote, Google Images results, or are images we generated with OpenAI GPT-4.

[2]The current version of this paper is v1.1-2023.12 (In the format of '[major].[minor]-YYYY.MM').

Contents

1 Overview

1.1 Introduction

We still face many challenges in developing autonomous robotic systems that can operate in and adapt to different environments. Previous robotic perception systems that leverage conventional deep learning methods usually require a large amount of labelled data to train the supervised learning models [1–3]; meanwhile, the crowdsourced labelling processes for building these large datasets remains rather expensive. Moreover, due to the limited generalization ability of classical supervised learning approaches, the trained models usually require carefully-designed domain adaptation techniques to deploy these models to specific scenes or tasks [4, 5], which often require further steps of data-collection and labelling. Similarly, classical robot planning and control methods often require carefully modelling the world, the ego-agent's dynamics, and/or other agents' behavior [6–8]. These models are built for each individual environment or task and often need to be rebuilt as changes occur, exposing their limited transferability [8]; in fact, in many cases, building an effective model can be either too expensive or intractable. Although deep (reinforcement) learning-based motion planning [9, 10] and control methods [11–14] could help mitigate these problems, they also still suffer from distribution shifts and reductions in generalizability [15, 16].

Concurrent to the challenges faced in building generalizable robotic systems, we notice significant advances in the fields of Natural Language Processing (NLP) and Computer Vision (CV)—with the introduction of Large Language Models (LLMs) [17] for NLP, the introduction of high-fidelity image generation with diffusion models [18, 19], and zero-shot/few-shot generalization of CV tasks with large-capacity vision models and Vision Language Models (VLMs) [20–22]. Coined "foundation models" [23], or simply Large Pre-Trained Models (LPTMS), these large-capacity vision and language models have also been applied in the field of robotics [24–26], with the potential for endowing robotic systems with open-world perception, task planning, and even motion control capabilities. Beyond just applying existing vision and/or language foundation models in robotics, we also see considerable potential for the development of more robotics-specific models, e.g., the action model for manipulation [27, 28] or motion planning model for navigation [29]. These robotics foundation models show great generalization ability across different tasks and even embodiments. Vision/language foundation models have also been applied directly to robotic tasks [30, 31], showing the possibility of fusing different robotic modules into a single unified model.

Although we see promising applications of vision and language foundation models to robotic tasks, and the development of novel robotics foundation models, many challenges in robotics out of reach. From a practical deployment perspective, models are often not reproducible, lack multi-embodiment generalization, or fail to accurately capture what is feasible (or admissible) in the environment. Furthermore, most publications leverage transformer-based architectures and focus on semantic perception of objects and scenes, task-level planning, or control [28]; other components of a robotic system, which could benefit from cross-domain generalization capabilities, are under-explored—e.g., foundation models for world dynamics or foundation models that can perform symbolic reasoning. Finally, we would like to highlight the need for more large-scale real-world data as well as high-fidelity simulators that feature diverse robotics tasks.

In this paper, we investigate where foundation models are leveraged within robotics, am aim to understand how foundation models could help mitigate core robotics challenges. We use the term "***foundation models for robotics***" to include two distinct aspects: (1) the application of existing (mainly) vision and language models **to** robotics, largely through zero-shot and in-context learning; and (2) developing and leveraging **robotics foundation models** specially for robotic tasks by using robot-generated data. We summarize the methodologies of foundation models for robotics papers and conduct a meta-analysis of the experimental results of the papers we surveyed. A summary of the major components of this paper in Figure 1.

The overall structure of this paper is formulated as in Figure 2. In Section 2, we provide a brief introduction to robotics research before the foundation model era and discuss the basics of foundation models. In Section 3, we enumerate challenges in robotic research and discuss how foundation models might mitigate these challenges. In Section 4, we summarize the current research status quo of foundation models in robotics. Finally, in Section 6 we offer potential research directions which are likely to have a high impact on this research intersection.

1.2 Related Survey Papers

Recently, with the popularity of foundation models, there are various survey papers on vision and language foundation models that are worth mentioning [32–35]. These survey papers cover foundation models, including Vision Foundation Models

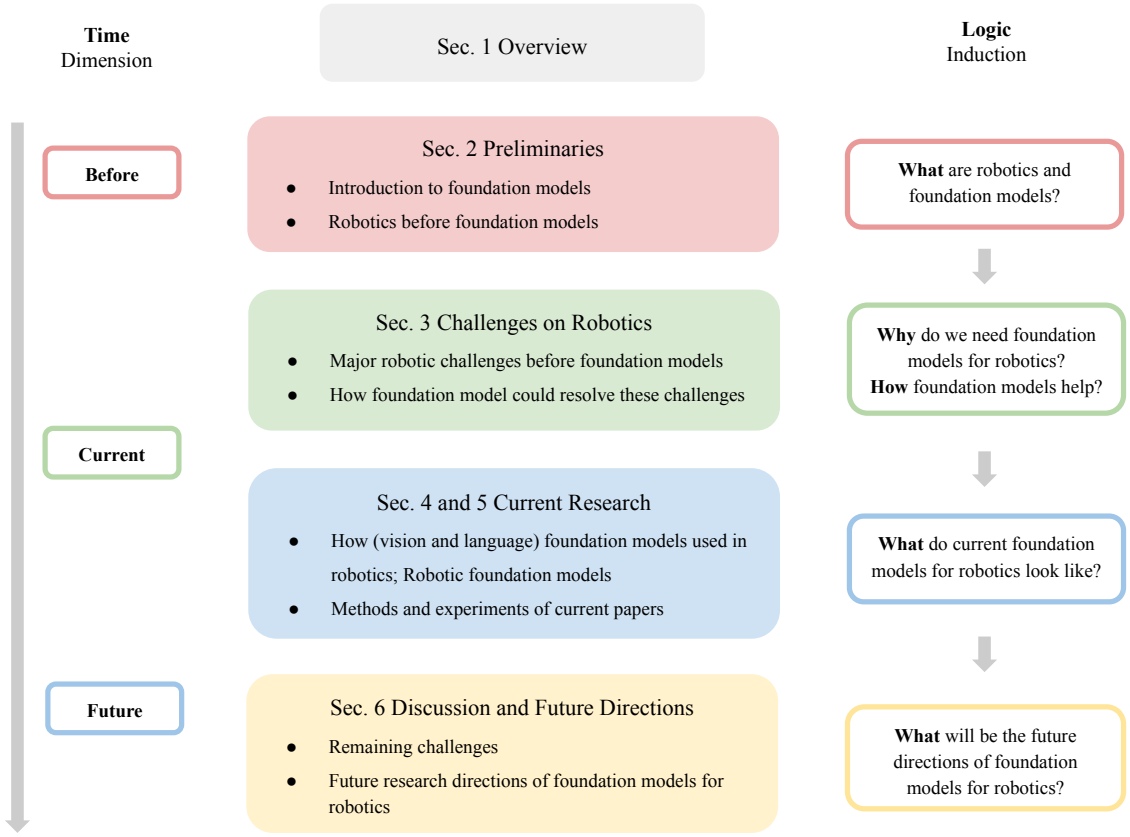


Figure 2: Overall structure of this survey paper. The left side shows the time dimension governing the emergence of foundation models, and the right side shows the logical induction, with respect to applying and developing foundation models for robotics. Sections 2 and 3 answer the "what" questions: what is robotics, what are the challenges in robotic problems, and what are foundation models. Section 4 and 5 deal with the "why" and "how" questions: why do we need foundation models in robotics, how can foundation models be applied in robotics, and how do foundation models specially designed for robotics work. Section 6 closes with existing work, and posits what future models might look like.

(VFMs) [36, 37], Large Language Models (LLMs) [34], Vision-Language Models (VLMs) [38, 39], and Visual Content Generation Models (VGMs) [32]. Since foundation models for robotics are still relatively nascent areas, there are not so many existing survey papers combining foundation models and robotics, perhaps the most relevant survey papers are [35, 40–43], however, there are still significant differences between those papers and ours, for instance: Yang *et al.* [35] and Wang *et al.* [40] focus on broadly-defined autonomous agents, instead of physical robots; Lin *et al.* [41] focus on LLMs for navigation; the connection between foundation models and robotics is limited in [42]. Compared with [43], we propose more breakdown of current research methodologies, provide detailed analysis of the experiments, and also focus on how foundation models could resolve the typical robotic challenges. Concurrently, Firoozi *et. al.* [44] conducted a survey regarding foundation models in robotics. Both their and our works shed lights on the opportunities and challenges of using foundation models in robotics, and identifies key pitfalls to scale them further. Their work focuses on how foundation models contribute to improving robot capabilities, and challenges going forward. Comparatively, our survey attempts to taxonomize robotic capabilities together with foundation models to advance those capabilities. We also propose a dichotomy between robotic(-first) foundation models and other foundation models used in robotics, and provides a thorough meta-analysis of the papers we survey.

In this paper, we provide a survey that includes existing unimodal and multimodal foundation models applied in robotics, as well as all forms of robotics foundation models in various robotics tasks as we know of. We also narrowed the scope of papers being surveyed to only those with experiments on real physical robotics, in high-fidelity simulation environments, or using real robotics datasets. We believe that, by doing so, it could help us understand the power of foundation models in the real world robotic applications.

2 Preliminaries

In this section, we walk through the preliminaries to help readers better understand the contents of this paper. Since we focus on foundation models centered around robotics, we will first introduce the basics of robotics and the current state of the art. These preliminaries will focus on the methods before foundation models were applied in robotics. For the ease of organization, we introduce robotic modules based on their functionalities, e.g., perception 2.1.1, planning 2.1.2 and control 2.1.3. We note that, although we introduce these modules separately, the boundaries between these modules are often blurred [12, 45]: these modules are often synergistically connected, facilitating end-to-end differentiability—allowing gradients to flow across different modules, especially in learning-based approaches. The second focus of this section will be to provide an introduction to foundation models, mainly situated in the fields of NLP and CV; these models include: LLMs, VLMs, vision foundation models, as well as text-conditioned image generation models.

2.1 Ingredients of a Robotic System

2.1.1 Robot Perception

Robots require perceptual mechanisms, in order to extract semantic knowledge from raw sensor observations, establish state representations, and enact reasoning in their operating environments. Different from typical computer vision systems, robotic perception emphasizes real-time capability, the use of multiple modalities (RGB, depth, LiDAR, IMU, tactile, etc.), the coupling with other robotic system modules (decision making, planning, control), and grounding with the embodiment and the environment [46, 47].

Passive Perception The most common use-case for typical computer vision algorithms in a robot perception system is for scene understanding. Here, the goal is to extract insights about the semantic and geometric properties of an environment by processing visual signals (e.g., 2D image data, RADAR information, LiDAR/RGB-D point clouds), perhaps to perform specific tasks like object detection & tracking, semantic segmentation, pose-estimation, novel view synthesis, or scene reconstruction [48–52]. However, the problem with the dominant learning-based approaches is that they primarily rely on large amounts of labeled data for training, where it is particularly challenging to obtain large-scale labels. Furthermore, these approaches tend to break down in out-of-distribution scenarios, rendering them too brittle for extensive deployment. As discussed in Section 2.2.1, this problem has been largely alleviated with the impressive open-set capabilities of visual foundation models [22, 53].

State Estimation State estimation is the challenging problem of estimating the poses or velocities of robots based on sensor measurements. The task of Simultaneous Localization and Mapping (SLAM) seamlessly integrates the pose estimation and mapping problems together. State estimation and SLAM can be addressed using various sensor modalities, as evidenced by the collection of vision-based approaches [54–59], LiDAR-based techniques [60–62], methods leveraging Inertial Measurement Units (IMU) [63, 64], and sensor fusion methods involving multiple sensors [65–69]. While traditional approaches typically rely on rigorous geometry-based solutions, more recently there has been a growing interest in learning-based approaches, which leverage supervised [70, 71] and self-supervised [72–75] methods. These learning-based methods have demonstrated their ability to provide accurate pose-tracking results and, in some cases, achieve dense reconstruction even without the use of depth sensors.

Active Perception The previously introduced approaches give a robot the ability to perceive the environment only in a passive manner, where information gain or decision-making does not play a role in how the perception system evolves, temporally. Since robots move and often interact with the environment, the robot should also be able to perceive the environment in an active manner. Prior approaches to the active perception problem have looked to the lens of interacting with the environment [76] or by changing the viewing direction to obtain more visual information [77–79].

2.1.2 Robot Decision-making and Planning

Classical Planning Planning for robotics is a process of organizing a set of actions that a robot should execute, given a model of itself and the world, to reach desired states and minimize cumulative costs when executing those actions. Motion planning aims to find a collision-free path to reach desired states. Search-based planning [80–84] computes robots' trajectories based on

discrete representations of problems, taking advantage of heuristics and graphs. Another major area, sampling-based planning [85–90], seeks to randomly sample points in configuration spaces, find paths to desired states by connecting nearby points or incrementally generating next states, and is well-suited for high-dimensional planning and continuous spaces.

Task planning [91] deals with discrete domains with tractable and compact representations for large state-spaces, usually exploiting structural properties of the domains and object-level abstractions. The approaches for compact representations include factoring the state representations into a set of smaller state variables, use of a set of preconditions that specify the states in which robots can execute actions, and object-oriented abstraction for symbolic reasoning [92].

Learning-based planning There exist newer works in planning that use reinforcement learning [93, 94] and formulate motion-planning as an end-to-end problem [9, 10]. In navigation, `ReViND` [95] and `FastRLAP` [94] use offline reinforcement learning to learn the planning policy for visual navigation. By optimizing a value function from a static dataset, robots could learn driving behaviors in a short period of time [94]. In addition, the advantage of utilizing reward re-labeling allows the robot to earn different navigation behavior according to the reward specified by the user [95]. Learning for task planning includes recent works on integrating reinforcement learning with task planning [96, 97] to improve adaptability in dynamic environments and generate plans, and works on learning symbolic abstract model and representations for task planning [98, 99].

2.1.3 Robot Action Generation

Classical Control Low-level action control, achieved via direct actuation or motor control, is the last step in most robotics stacks. This part of the stack is usually dependent on the exact platform and often incorporates dynamics and actuator constraints, thereby ensuring the feasibility of the generated action by keeping the robot within its operational envelope. While the control input is usually in continuous space, motion primitives are often used to provide a discrete set of actions for ease of interfacing with higher-level decision loops. Arguably, PID control loops are the most widely used lower-level control structures for robotic systems. When a cost function is available, optimization-based methods, a.k.a. optimal control, such as Model Predictive Control (MPC) and its variants [100–103] are often used to generate action sequences in a receding-horizon setup. The Model Predictive Path Integral (MPPI) controller [104], a variant of MPC, is widely used in its sampling formulation on learned cost maps.

Learning-based Control Applying imitation learning [105] or reinforcement learning [106] in robotic control has been studied for decades. With the success of deep learning [107] and deep reinforcement learning [108, 109], we see this line of research getting a good number of success stories in recent years [12, 13, 110–113].

Imitation learning aims to learn a control policy by imitating demonstrations from some expert, which could be implicit in a dataset of trajectories. It can be in the form of supervised learning which directly learns actions from expert demonstrations [114], inverse reinforcement learning [115, 116] which learns reward functions, and adversarial imitation learning which learns the policy with generative adversarial networks [117, 118]. Imitation learning is widely used in various robot control applications, including: urban driving [101], high-speed car racing [119], autonomous drone acrobatics [110], learning locomotion skills by imitating animals [11], and quadrupedal agile skills via adversarial imitation learning [120].

Reinforcement learning (RL) [121] is typically leveraged in the context of a Markov Decision Process, in order to learn and optimize a control policy via accumulated rewards. Different from optimal control-based methods, RL methods may not require dynamics models. Many existing works in RL for robotics follow a model-free learning paradigm, wherein a policy learns to directly map sensor observations (e.g., images [12, 111, 122], proprioception [13, 113, 123], or both [124]) to generate actions. Model-free approaches usually have the drawback of being sample-inefficient. Model-based RL methods [125] provide a viable solution by learning a world dynamics model and then planning or learning the control policy. These world dynamics model can be in the form of visual observations [126, 127] or dynamics models based on proprioception [14]. However, aforementioned online RL methods in real-world may give rise to safely concerns [95]. Offline reinforcement learning approaches [95, 128, 129] attempt to remedy this problem, since the learning is based only on offline datasets. We have already seen a few works which apply offline RL in robotics, such as in visual navigation [93], high-speed ground vehicle driving [94], and manipulation [130, 131].

2.2 Introduction to Foundation Models

From the definition in [23], a foundation model is any model that is trained on broad data (generally using self-supervision at scale) that can be transferred or adapted (e.g., fine-tuned) to a wide range of downstream tasks. Existing successful foundation models are mainly from CV and NLP areas, e.g. (ranked from single modalities to multiple modalities): Vision Foundation Models (VFMs; Section 2.2.1), Large Language Models (LLMs; Section 2.2.3), Vision-Language Models (VLMs; Section 2.2.4), and Large Multimodal Models (LMMs; Section 2.2.5). Recently, we also see the rise of foundation models that are specifically designed for robotics tasks and trained on robotics data ("robotics foundation models"): this particular topic will be introduced, later, in Section 4.2.

2.2.1 Vision Foundation Models (VFMs)

Alongside the advent of LLMs and VLMs, several Vision-based Foundation Models (VFMs) have been proposed [20–22, 132]. Owing to their impressive domain-invariance and semantic properties at the pixel- and object-level [133–137], these vision foundation models have been widely adopted for downstream passive perception tasks. Furthermore, these major advances have been enabled either through self-supervision [133] and/or large-scale data curation [21, 22].

The family of self-supervised VFMs can be broadly organized into the following three subclasses: (1) Joint-Embedding Predictive Architectures (JEPA; [138]), (2) Contrastive Learning-based methods [53, 139], (3) and Masked Autoencoder (MAE; [132]) JEPAs employ a Bootstrap Your Own Latent (BYOL; [140]) style of self-supervision technique, where the primary supervisory signal is to predict similar embeddings across different augmentations of an image. Amongst the JEPA methods, the most notable ones are `DINO` [20], `DINOv2` [22], `I-JEPA` [141] and `MC-JEPA` [142]. Recent explorations have shown that these joint-embedding-based approaches capture longer-range global patterns and shape-oriented features [133, 134]. On the other hand, contrastive learning-based methods leverage the weak supervision from multimodal data to learn a common latent space across different modalities. Notable methods include `CLIP` [53], which uses large-scale image-caption pairs. In addition to these two classes, MAEs [132] make up another class of models, trained to reconstruct masked inputs as a pretext task. Explorations have shown that these models capture local token-level semantic context, leading to their wide popularity for dense prediction problems such as semantic segmentation [133, 134].

Two notable VFMs that have been enabled by the careful curation of large datasets are the Segment Anything Model (`SAM` [21]) and `DINOv2` [22]. `SAM` leveraged an iterative model prediction-based curation process to obtain 1 billion semantic segmentation masks for supervised learning. It has been showcased that the `SAM` models, trained on this large-scale curated data, show impressive instance segmentation performance across a wide range of domains. Similarly, `DINOv2` [22] is a self-supervised model trained using model prediction-based curated data comprising 142 million images. It has been showcased that this large-scale self-supervision on curated data enables `DINOv2` to perform better than task-specifically trained models and contrastive zero-shot models such as `CLIP`, while showcasing impressive semantic consistency.

2.2.2 Visual Content Generation Models (VGMs)

Text-conditioned image generation models have achieved great attention recently due to their astonishing ability to generate novel, high-fidelity images directly from language prompts, thanks to the progress in diffusion models [143]. `GLIDE` [144] is a text-conditioned diffusion model with both `CLIP` guidance and classifier-free guidance. `DALLE-2` [18] proposes a two-stage diffusion model that consists of a prior that generates a `CLIP` image embedding given a text caption and a decoder that generates an image conditioned on the encoded image embedding. `IMAGEN` [19] is another text-conditioned diffusion model with classifier-free guidance. Different from previous approaches, it proposes dynamic thresholding to generate more photorealistic images and a U-Net structure to make the training more efficient. We name this type of foundation model as Visual Content Generation Models (VGMs) in this paper for convenience.

2.2.3 Large Language Models (LLMs)

A Large Language Model (LLM) is a type of language model notable for its ability to handle a variety of language tasks with minimal task-specific training data, setting it apart from conventional AI models [145]. The term *large* refers to both the model size and dataset size. Moreover, *language* signifies that the models are trained on an internet scale with a single modality, which is text. The key development in LLMs was the introduction of the transformer architecture, which allows for the efficient

training of large-scale data due to the highly parallel nature of transformers, making the processing of extended text sequences more efficient. Two lines of work build upon the transformer architecture: the Generative Pre-trained Transformer (GPT) series [17, 146] and the Bidirectional Encoder Representations from Transformers (BERT) family [147]. GPT is trained as a decoder, with the task of predicting the next word in a sequence, whereas BERT is trained as an encoder, focusing on understanding the contextual relationships between sentences. However, according to Yang *et al.* [148], the BERT family has not seen significant advancements since 2021, with all current state-of-the-art LLMs building upon the GPT structure. Another technique that has contributed to the achievements of models like the `GPT-3.5` model (ChatGPT) is the paradigm of Reinforcement Learning from Human Feedback (RLHF) [149]. This technique aligns the transformer's output with human preferences learned through inverse reinforcement learning, enabling the model to produce more human-like and fluent sentences.

However, the limitations of LLMs are still significant, particularly in their tendency to hallucinate to produce plausible outputs. In addition, due to the limitations of being trained only on a single text modality, LLMs lack references to real-world objects and, therefore, cannot accurately answer questions such as, "*Can I put this watermelon into the blender?*". This problem of lacking spatial grounding is especially important for the application of LLMs to Robotics. As a result, more recent research has started to integrate additional sources of information, such as vision, into transformer architectures.

2.2.4 Vision-Language Models (VLMs)

Vision-language models (VLMs) represent another significant advancement in generative models. These models excel in tasks that require understanding both visual content and language, such as open-set image classification [53], object detection [150], and segmentation [151], Visual Question Answering (VQA) [152], etc. These models leverage large-scale datasets and sophisticated neural network architectures, typically variants of transformers, to learn correlations between images and their textual descriptions or queries. This approach enables them to perform a range of tasks without task-specific training, showcasing impressive generalization abilities. They can be broadly categorized into two main groups based on their pre-training methods: contrastive learning models and generative pre-training models.

Contrastive learning based models, like `CLIP` [53] and `ALIGN` [153], are trained to understand the correspondence between images and texts by bringing the representations of matching image-text pairs closer in the embedding space while pushing non-matching pairs apart. `CLIP`, for instance, excels in a wide range of visual classification tasks with its ability to understand nuanced textual descriptions and their corresponding images. `ALIGN`, similarly, focuses on aligning large-scale image-text pairs, significantly improving performance on tasks like image captioning and visual question answering. *Generative pre-training models*, like `ViLBERT` [152] and `VL-BEiT` [154] take similar training approach as in LLMs 2.2.3.

Combing the strength of the both methods, `Flamingo` [155] incorporates a large frozen language model, retaining the in-context few-shot learning ability that is inherent to the pre-trained language model. Conversely, `GIT` [156] employs a large contrastively pre-trained image encoder, accompanied by a relatively small text decoder. Both `Flamingo` and `GIT` commence by pre-training an image encoder through contrastive learning, then subsequently undertaking generative pre-training.

More recently, `BLIP` [157] and `BLIP-2` [158] have emerged as a transformative model in VLMs, introducing a curriculum learning strategy that bootstraps from simpler to more complex tasks, significantly enhancing performance in tasks like image captioning and visual question answering. The latest GPT iteration, `GPT-4` [159], introduces the capability to process both textual and visual inputs. However, as of the last update, the technical details and the extent of these new multimodal capabilities are not released yet. Collectively, these models exemplify the rapid advancements in VLMs, each contributing to the robustness and adaptability of multimodal systems in understanding and in generating human-like responses based on visual data.

2.2.5 Large Multimodal Models (LMMs)

The combination of vision and language modalities reveal the huge potential of self-supervised learning. So it is natural to extend beyond vision and language to develop new types of foundation models with even more modalities. These models are coined Large Multimodal Models (LMMs). The additional modalities are rich and diverse, as in models that combine image, text, depth, thermal, and audio, e.g., `ImageBind` [139]; models combining text, image, video and audio, e.g., `NExT-GPT` [160] and `Audio-GPT` [161]; models combining language text and audio, e.g., `SpeechGPT` [162]; and models combining point cloud with vision and language, e.g., `ULIP` [163]. These LMMs use distinct training methods, such as contrastive pre-training [139] or fine-tuning LLM by learning input/output projections [160]. The contrastive learning

methods shown in these LMMs are fairly similar to LLMs, in the sense that embeddings of different modalities are aligned with each other. These multimodal contrastive learning-based methods enable cross-modal retrieval and bring in more interesting applications such as audio to image generation, etc.

3 Challenges in Robotics

In this section, we summarize five core challenges that various modules in a typical robotic system face, each detailed in the following subsections. Whereas similar challenges have been discussed in prior literature (Section 1.2), this section mainly focuses on the challenges that may potentially be solved by appropriately leveraging foundation models, given the evidence from current research results. We also depict the taxonomy in the section for easier review in Figure 3.

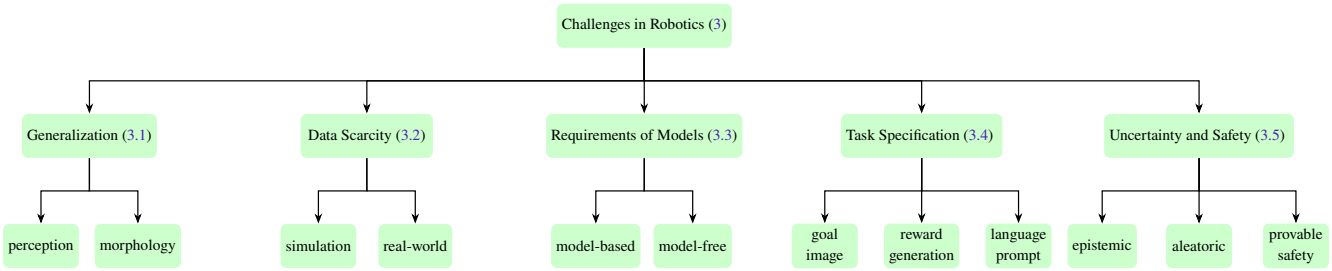


Figure 3: Taxonomy of the challenges in robotics that could be resolved by foundation models. We list five major challenges in the second level and some, but not all, of the keywords for each of these challenges.

3.1 Generalization

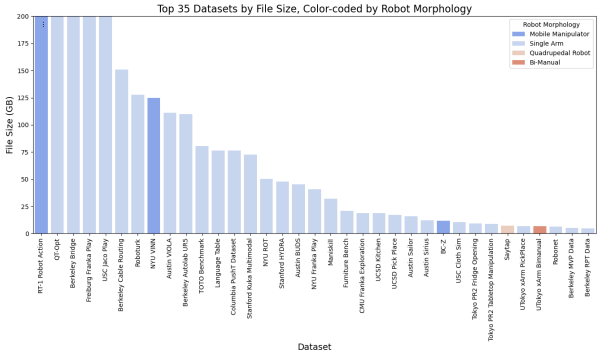
Robotics systems often struggle with accurate perception and understanding of their environment. Limitations in computer vision, object recognition, and semantic understanding made it difficult for robots to effectively interact with their surroundings. Traditional robotics systems often relied on analytic hand-crafted algorithms, making it challenging to adapt to new or unseen situations. They also lacked the ability to generalize their training from one task to another, further limiting their usefulness in real-world applications. This generalization ability is also reflected in terms of the generalization of planning and control in different tasks, environments, and robot morphologies. For example, specific hyperparameters for, e.g., classical motion planners and controllers need to be tuned for specific environments [102, 103, 164]; RL-based controllers are difficult to transfer across different tasks and environments [47, 165]. In addition, due to differences in robotic hardware, it is also challenging to transfer models across different robot morphologies [166, 167]. By applying foundation models in robotics, the generalization problem is partially resolved, which will be discussed in the next Section 4. Further challenges, such as generalization across different robotic morphologies, remain demanding.

3.2 Data Scarcity

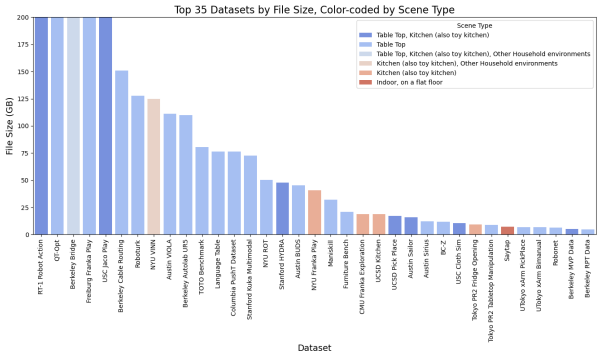
Data has always been the cornerstone of learning-based robotics methods. The need for large-scale, high-quality data is essential to develop reliable robotic models. Several endeavors have been attempted to collect large-scale datasets in the real world, including autonomous driving [1, 2, 168], robot manipulation trajectories [111, 112, 169], etc. Collecting robot data from human demonstration is expensive [27]. The diverse range of tasks and environments where robots are used even complicates the process of collecting adequate and extensive data in the real world. Moreover, gathering data in real-world settings can be problematic due to safety concerns [164]. To overcome these challenges, many works [170–175] attempt generating synthetic data in simulated environments. These simulations offer realistic virtual worlds where robots can learn and apply their skills to real-life scenarios. Simulations also allow for domain randomization, as well as the potential to update the parameters of the simulator to better match the real world physics [164], helping robots to develop versatile policies. However, these simulated environments still have their limits, particularly in the diversity of objects, making it difficult to apply the learned skills directly to real-world situations. Collecting real-world robotic data with a scale comparable to the internet-scale image/text data used to train foundation models is especially challenging. One promising approach is collaborative data collection across different laboratories and robot types [176], as shown in Figure 4a. However, an in-depth

analysis of the Open-X Embodiment Dataset reveals certain limitations regarding data type availability. Primarily, the robot morphology utilized for data collection is restrictive; out of the top 35 datasets, 30 are dedicated to single-arm manipulation tasks. Only one dataset pertains to quadruped locomotion, and a single dataset addresses bi-manual tasks. Secondly, the predominant scene type for these manipulation tasks is tabletop setups, often employing toy kitchen objects. Such objects come with inherent assumptions including rigidity and negligible weight, which may not accurately represent a wider range of real-world scenarios. Thirdly, our examination of data collection methods indicates a predominance of human expert involvement, predominantly through virtual reality (VR) or haptic devices. This reliance on human expertise highlights the challenges in acquiring high-quality data and suggests that significant human supervision is required. For instance, the RT-1 Robot Action dataset necessitated a collection period of 17 months, underscoring the extensive effort and time commitment needed for data accumulation with human involvement.

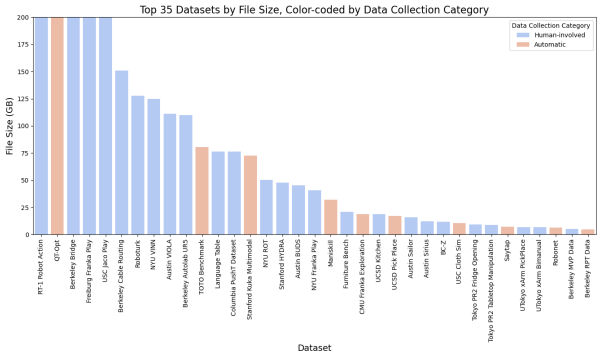
(a) Joint Effort in Real-World Data Collection



(b) Dataset Morphology Distribution



(c) Scene Type Classification



(d) Data Collection Methods

Figure 4: Comprehensive visualizations of the Open-X Embodiment Dataset encompassing data collection methods, robot morphologies, and scene types.

3.3 Requirements of Models and Primitives

Classical planning and control methods usually require carefully engineered models of the environment and the robot. Optimal control methods require good dynamics models (i.e., world transition models) [8, 177]; motion planning requires a map of the environment [178], the states of the objects robots interact with [179], or a set of pre-defined motion primitives [180]; task planning requires pre-computed object classes and pre-defined rules [92], etc. Previous learning-based methods (e.g., imitation and reinforcement learning) train policies in an end-to-end manner that directly gets control outputs from sensory inputs [112], avoiding building and using models. These methods partially solve this problem of relying on explicit models, but they often struggle to generalize across different environments and tasks. This raises two problems: (1) How can we learn model-agnostic policies that can generalize well? Or, alternatively, (2) How can we learn good world models so that we can apply classical model-based approaches? We see some recent works that aim to resolve these problems using foundation models (especially in a model-free manner), which will be systematically discussed in Section 4. However, the call for world models for robotics remains an intriguing frontier, which will be discussed in Section 6.

3.4 Task Specifications

Understanding the task specification and grounding it in the robot's current understanding of the world is a critical challenge to obtaining generalist agents. Often, these task specifications are provided by users with limited understanding of the limitations on the robot's cognitive and physical capabilities. This not only raises questions about what the best practices are for providing these task specifications, but also about the naturalness and ease of crafting these specifications. Understanding and resolving ambiguity in task specifications, conditioned on the robot's understanding of its own capabilities, is also challenging. Foundation models, again, are a promising solution for this challenge: task specification can be formulated as language prompts [24, 27, 28], goal images [181], rewards for policy learning [26, 182], etc.

3.5 Uncertainty and Safety

One of the critical challenges in deploying robots in the real world comes from dealing with the uncertainty inherent in the environments and task specifications. Uncertainty, based on its source, can be characterized either as epistemic (uncertainty caused by a lack of knowledge) or aleatoric (noise inherent in the environment). Epistemic uncertainty often manifests as out-of-distribution errors when the robot encounters unfamiliar situations in the test distribution. While the adoption of learning-based techniques for decision-making in high-risk safety-critical fields has prompted efforts in uncertainty quantification (UQ) and mitigation [183], out-of-distribution detection, explainability, interpretability, and vulnerability to adversarial attacks remain open challenges. Uncertainty quantification can be prohibitively expensive and may lead to sub-optimal downstream task performance [184]. Given the large-scale over-parameterized nature of foundation models, providing UQ methods that preserve the training recipes with minimal changes to the underlying architecture are critical in achieving the scalability without sacrificing the generalizability of these models. Designing robots that can provide reliable confidence estimates on their actions and in turn intelligently ask for clarification feedback remains an unsolved challenge [185]. Conformal predictions [186] provide a distribution-free way of generating statistically rigorous uncertainty sets for any black-box model and have been demonstrated in VLN tasks for robotics [187].

In its traditional setting, provable safety in robotics [188, 189] refers to a set of control techniques that provide theoretical guarantees on safety bounds for robots. Control Barrier Functions [190], reachability analysis [122, 191] and runtime monitoring via logic specifications [192] are well-known techniques in ensuring robot safety with bounded disturbances. Recent works have explored the use of these techniques to ensure safety of the robot [193]. While these contributions have led to improved safety, solutions often result in sub-optimal behavior and impede robot learning in the wild [194]. Thus, despite recent advances, endowing robots with the ability to learn from experience to fine-tune their policies while remaining safe in novel environments still remains an open problem.

4 Review of Current Research Methodologies

In this section, we summarize the current research methodologies of foundation models for Robotics. In Section 4.1, we mainly discuss how foundation models are used in robotics in two categories: **Foundation Models used in Robotics** and **Robotics Foundation Models** (RFMs). For Foundation Models used in Robotics, we mainly highlight applications of **vision and language** foundation models used in a *zero-shot* manner, meaning no additional fine-tuning or training is conducted. In Section 4.2, however, we mainly focus on Robotics Foundation Models, wherein these approaches may warm-start models with vision-language pre-trained initialization and/or directly train the models on **robotics datasets**. Figure 5 shows the detailed taxonomy of this section.

As introduced in the Section 2 (Preliminaries), a typical robotic system consists of perception, planning, and control modules. In this section, we review the methods presented in these papers following this classification method. Here, we combine motion planning and control into one piece—action generation and treat motion planning modules as higher-level and control as lower-level action generation. It is important to notice that although most of the works use foundation models in different functionality modules of the robotic systems, we will classify these papers into categories based on the module to which the paper contributes the most. There are, however, certain applications of the vision and language foundation models that go across these robotics modules, e.g., grounding of these models in robotics, and generating data from LLMs and VLMs. Given the autoregressive nature of current LLMs, they often grapple with extended horizon tasks. Thus, we also delve into

advanced prompting methods proposed in the literature to ameliorate this limitation and enhance planning power. We list these applications in sections 4.1.4, 4.1.5 and 4.1.6, as a different perspective to analyze these works.

We find that works in Section 4.1 typically follow a modular strategy, in applying vision and language foundation models to serve a single robot functionality, e.g., applying VLMs as open-set robot perception modules which are then "plugged in" to work alongside motion planners and controllers [25], downstream. Since such foundation models are applied in a zero-shot manner, there are no gradients flowing between the module in which the foundation models are applied and the other modules in the robotic system. Conversely, works in Section 4.2 mostly follow an end-to-end differentiability paradigm, which blurs the boundary of the typical robotics modules in methods (described in Section 4.1; e.g., perception and control [27, 195]), with some robotics foundation models even providing a unified model to perform different robot functions [30, 31].

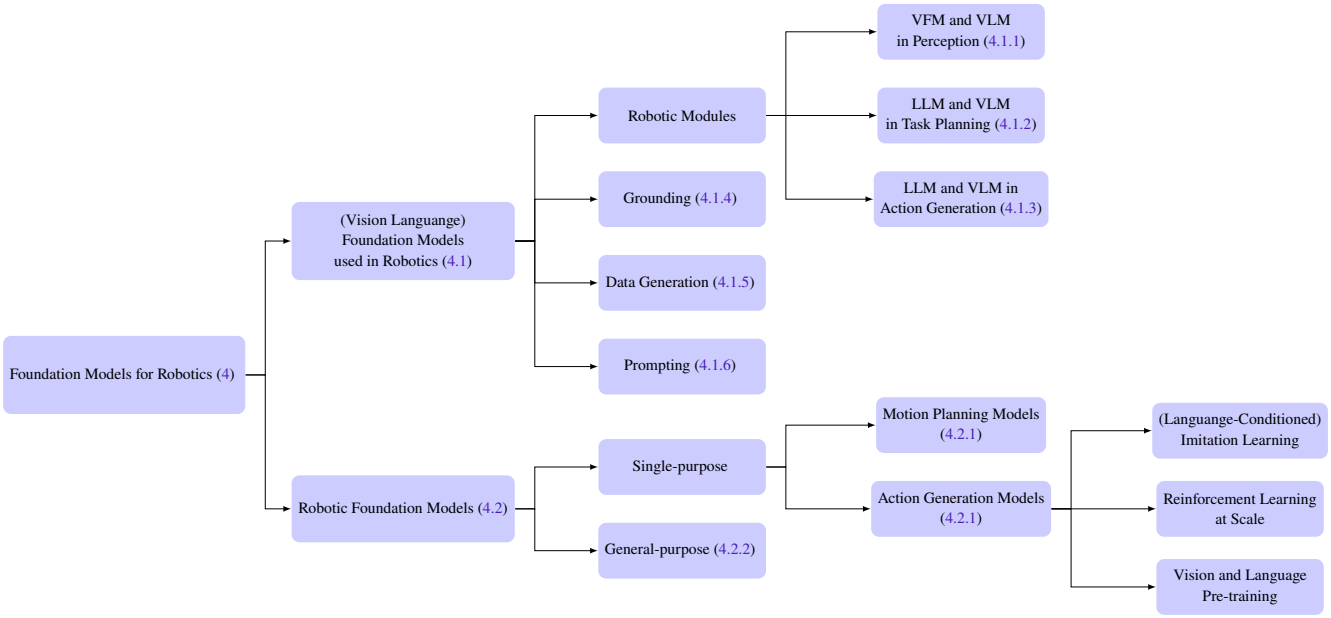


Figure 5: Conceptual Framework of Foundation Models in Robotics: The figure illustrates a structured taxonomy of foundational models, categorized into two primary segments: the application of existing foundation models (vision and language models) to robotics, and the development of robotic-specific foundation models. This includes distinctions between vision and language models used as perception tools, in planning, and in action, as well as the differentiation between single-purpose and general-purpose robot foundation models.

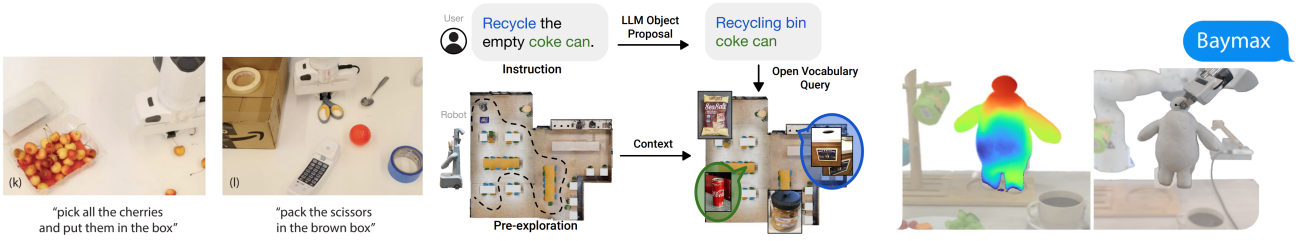
4.1 Foundation Models used in Robotics

In this section, we focus on *zero-shot* application of vision and language foundation models in robotics. This mainly includes zero-shot deployment of VLMs used in robotic perception, in context learning of LLMs for task-level and motion-level planning, as well as action-generation. We show a few representative works in Figure 6.

4.1.1 VFMs and VLMs in Robot Perception

Recently, the grounding of vision and language foundation models with geometric and object-centric representations of the world has enabled tremendous progress in context understanding, which is a vital requirement for robots to interact with the real world. We will thoroughly examine the application of VFMs and VLMs in robotic perception from various perspectives.

VFMs, VLMs for Object and Scene Representations The most straightforward application of VLMs in robotics is to leverage their ability to perform open-set object recognition and scene understanding in robotics-oriented downstream tasks, including semantic mapping and navigation [25, 201–204], manipulation [196–198, 205], etc. The methods proposed by these works share a common attribute: they attempt to extract *semantic* information (from the VLM) and *spatial* information (from other modules or sensor modalities) from objects and scenes that the robots interact with. This information is then used as representations in semantic maps of scenes or representations of objects.



(a) Object recognition: `CLIPort` [196] (b) Object recognition & localization: `NLMap` [25] (c) Feature distillation in NeRF: `F3RM` [197]

VLM used for: **Object and Scene Representation**

(d) Representation in policy learning (also applied feature distillation in Ge-NeRF): `GNFactor` [198] (e) State estimation: `FM-Loc` [199] (f) Interactive perception: `MOSAIC` [200]

VFM and VLM used for: **Policy learning, State estimation and Interactive perception**

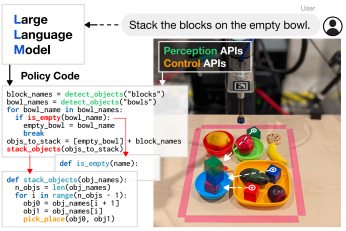
Figure 6: Using VLMs and VFMs for robotic perception, in various applications. Due to the limit of space and for the consistency of the figures, we only list part of the works we discussed in this figure.

For semantic mapping and/or navigation, `NLMap` [25] is an open-set and queryable scene representation to ground task plans from LLMs in surrounding scenes. The robot first explores the environment using frontier-based exploration to simultaneously build a map and extract class-agnostic regions of interest, which are then encoded by VLMs and aggregated to the map. Natural language instructions are then parsed by an LLM to search for the availability and locations of these objects in the scene representation map. `ConceptFusion` [203] builds open-set multimodal 3D maps from RGB-D inputs and features from foundation models, allowing queries from different modalities such as image, audio, text, and clicking interactions. It is shown that `ConceptFusion` can be applied for real-world robotics tasks, such as tabletop manipulation of novel objects and semantic navigation of autonomous vehicles. Similarly, `CLIP-Fields` [202] encodes RGB-D images of the scene to language-queryable latent representations as elements in a memory structure, that the robot policy can flexibly retrieve. `VLMap` [201] uses `LSeg` [151] to extract per-pixel representations to then fuse with depth information, in order to create a 3D map. This semantic 3D map is then down-projected to get a 2D map with the per-pixel embedding; these embeddings can then be matched with the language embedding from `LSeg` to obtain the per-pixel semantic mask for the 2D map. As for applying VLMs in topological graphs for visual navigation, `LM-Nav` [204] is a good example: it uses an LLM to extract landmarks used in navigation from natural language instructions. These landmark descriptions, as well as image observations, are then grounded in a pre-built graph via a VLM. Then, a planning module is used to navigate the robot to the specified landmarks.

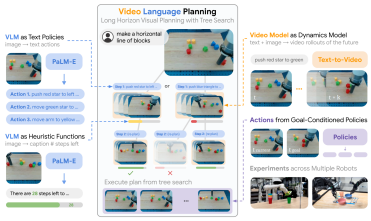
Most of the previous works discussed above utilize only 2D representation of the objects and environment. To enrich the representation of foundation models in 3D space, `F3RM` [197] and `GNFactor` [198] distill 2D foundation model features into 3D space, by combining with NeRF and generalizable NeRF. In addition, `GNFactor` [198] also apply these distilled features in policy learning. `Act3D` [206] takes a similar approach but build 3D feature field via sensed depth.

VLMs for State Estimation and Localization Beyond context understanding, a few approaches explore the use of open-vocabulary properties of VLMs for state estimation [199, 199, 207–209]. Two such approaches, `LEXIS` [207] and `FM-Loc` [199], explore the use of `CLIP` [53] features to perform indoor localization and mapping. In particular, `FM-Loc` [199] leverages the vision-language grounding offered by `CLIP` and `GPT-3` to detect objects and room labels of a query image, then uses that semantic information to match it with reference images. Similarly, `LEXIS` [207] builds a real-time topological SLAM graph where `CLIP` features are associated with graph nodes, enabling room-level scene recognition. Although these approaches display the potential of vision-language features for indoor place recognition, they do not explore the broad applicability of foundation model features. In this context, `AnyLoc` [208] explored the properties of dense foundation model

(a) Task plan generation via LLM and grounding of visual states, `SayCan` [24]



(b) Task planning via code generation LLM: `Code as policy` [216]



(c) VLMs for task planning and heuristic function, plus a video dynamics model: `VLP` [217]

Figure 7: Examples of LLMs and VLMs used in task-level planning

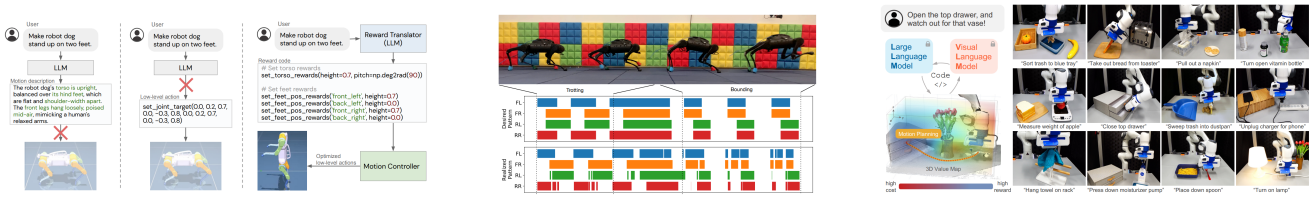
features and combined them with unsupervised feature-aggregation techniques to achieve state-of-the-art place recognition, by large margin—anywhere, anytime, and under any view—showcasing broad applicability of self-supervised foundation model features for SLAM.

VLMs for Interactive Perception Several works consider the notion of enabling robots to leverage the process of interactive perception, for extrapolating implicit knowledge about object properties in order to obtain performance improvements on downstream interactive robot learning tasks [166, 167, 200, 210–215]. This process of interactive perception is often modeled after the way in which human infants first learn about the physical world—i.e., through interaction, and by learning representations of object concepts, such as weight and hardness, from the sensory information (haptic, auditory, visual) that is generated from those physical exploratory actions (e.g., grasping, lifting, dropping, pushing) on objects with diverse properties. In particular, `MOSAIC` [200] leverages LMMs to expedite the acquisition of unified multi-sensory object property representations; the authors show competitive performances of their framework in category recognition and ambiguous target-object fetch tasks, despite the presence of distractor objects, under zero-shot transfer conditions.

4.1.2 LLMs and VLMs in Task Planning

The planning community in Robotics has always harbored aspirations of a model capable of generalizing across diverse tasks and environments, with minimal demonstrations for robotic tasks. Given the demonstrated prowess of vision and language foundation models in intricate reasoning and contextual generalization, it is a natural progression for the robotics community to consider the application of foundation models to robotic planning problems. This section organizes works based on the granularity of planning, delineating between task-level and motion-level planning. We will mainly introduce task-level planning in this part and leave motion-planning to the next part, together with action generation (Section 4.1.3).

Task-level planning is to divide a complicated task into small actionable steps. In this case, we mainly talk about the agent planning on its own, in contrast to, e.g., using LLMs as a parser like Vision Language Navigation [218]. The agent needs to take intelligent sub-steps to reach the goal by interacting with the environment. `SayCan` [24] is a representative example of task-level planning: it uses LLMs to plan for a high-level task, e.g., "I spilled my drink, can you help?". Then it gives concrete task plans like going to the counter, finding a sponge, and so on. Similarly, `VLP` [217] aims to improve the long-horizon planning approach with an additional text-to-video dynamics model. These task-level planning methods do not have to worry about the precise execution of those sub-tasks in the environment, since they have the luxury of utilizing a set of pre-defined / pre-trained skills, then using LLMs to simply find proper ways to compose skills to achieve desired goals. There are more papers in this category, for example: `LM-ZSP` [219] introduce this task-level granularity as actionable steps; `Text2Motion` [220] uses similar ideas and augments the success rate of language based manipulation task. Previous methods typically generate task plans in the form of text. Some works like `ProgPrompt` [221], `Code as Policy` [216], `GenSim` [222], etc. obtain task plans in the form of code generation using LLMs. Using code as a high-level plan has the benefit of expressing functions or feedback loops that process perception outputs and parameterize control primitive APIs. In addition, it can describe the spatial position of an object accurately. This improved compositionality saves time in collecting more primitive skills. It also prescribes precise values (e.g., velocities) to ambiguous descriptions like 'faster' and 'to the left', depending on the context. Therefore, due to these benefits, code appears to be a more efficient and effective task-level planning language than natural



(a) Reward generation with LLM : L2R [226]

(b) Locomotion control via action pattern matching with LLM: SayTap [26]

(c) VLM for affordance function in motion planning: VoxPoser [227]

Figure 8: LLM used in motion planning and action generation.

language. Other form of planning techniques such as express the high-level plans in Planning Domain Definition Language (PDDL)[223] also showed significant improvement on LLMs planning power over long horizon tasks, more on this will be discussed in the Section 4.1.6.

In addition to use LLMs to direct generate plans, they are also used in searching and evaluating with external memory structure such as scene graph. SayPlan [224] employs 3D scene graph (3DSG) representations to manage the complexity of expansive environments. By exploiting hierarchical 3DSGs, LLMs can semantically search for relevant sub-graphs in multi-floor household environments, reducing the planning horizon and integrating classical path-planning to refine initial plans, iteratively. Reasoned Explorer [225] employs LLMs as evaluators to score each node in a 2D undirected graph. It uses this graph as a map to store both visited points and the frontiers' LLM evaluations. These external memories and incremental map-building approach breaks the context length limit of using LLMs to generate long plans, which scales LLM-based navigation agents to large environments. One thing to note is that, although task-level planning is agnostic to physical embodiment, it does require grounding to a specific robot's physical form (or "morphology") and environment when deployed; grounding techniques will be covered in Section 4.1.4.

4.1.3 LLMs and VLMs in Action Generation

Directly controlling a robot just by prompting off-the-shelf LLMs/VLMs can be challenging, perhaps even unachievable, without first fine-tuning these models with action data. Unlike high-level robot task planning, where LLMs are used for their ability to compose and combine different skills for task completion, individual actions, both high-level actions like waypoints, and low-level actions like joint angles are usually not semantically meaningful or compositional. The community is attempting to find a suitable interface to circumvent this problem. For motion planning in navigation tasks, ReasonedExplorer [225] and Not_Train_Dragon [228] propose such an interface: using LLMs as evaluators for the expanded frontiers, which are defined as potential waypoints for exploration (typically in a two-dimensional space); here, LLMs are tasked with scoring frontiers based on the similarity between the given observations and the intended goal. Similarly, VoxPoser [227] apply VLMs to obtain affordance function (called 3D value map in the original paper) used in motion planning.

Some papers investigate the use of LLMs to directly output lower-level actions. Prompt2Walk [229] uses LLMs to directly output joint angles through few-shot prompts, collected from the physical environment. It investigates whether LLMs can function as low-level controllers by learning in-context with environment feedback data (observation-action pairs). Saytap [26], introduces a novel concept of utilizing foot contact patterns as an action representation. In this model, the language model outputs a '0' for no contact and a '1' for contact with the floor, thereby enabling Large Language Models (LLMs) to generate zero-shot actionable commands for quadrupedal locomotion tasks such as jumping and jogging. However, the generalizability of these approaches to different robot morphologies remains in question, since they have only been tested on the quadruped platform. Instead, language to reward [230–232] in robotics [182, 226] is a more general approach than direct action generation through LLMs; these approaches involve using LLMs as generators to synthesize reward functions for reinforcement learning-based policies and thus are usually not confined by robotic embodiments [182]. The reward synthesizing approach with LLM can generate rewards which are hard for human to design, e.g., Eureka [182] demonstrates that it enables robots to learn dexterous pen-spinning task that were considered very hard using human reward design.

4.1.4 Grounding in Robotics

Alongside the above dimensions, an equally crucial aspect is the concept of "grounding". Grounding refers to the ability to associate contextual meaning with signals or symbols, such as the ability to tie a word to its manifestation in the world. Humans understand semantic concepts through both audio (words, tone) and visual signals (gestures, behaviors, body language). In the scope of this survey, grounding alludes to the process of aligning the abstract knowledge possessed by foundation models with the tangible, real-world specifics of robotics—ensuring that language-driven decisions correspond meaningfully with physical actions and environmental contexts. For instance, if we ask an LLM to generate a plan to find a pen in a *specific* house, without any environmental information, then this task is akin to a blind person reasoning about how to navigate in an unknown space, rendering the task nearly impossible to complete. Similarly, while an LLM may easily generate a plan to lift a chair by suggesting to 'grab the left handle with the left hand and the right handle with the right hand, then lift', this plan becomes impractical if the real-world embodiment of the model is a typical robot equipped with only one arm. Since grounding is a large field in itself, we do not attempt to cover all of it; instead, we will address four concepts as shown in the Figure 9: (1) *Grounding language to environments*; (2) *Grounding latent concepts to environments*; (3) *Grounding language to embodiments*; and (4) *Grounding latent concepts to embodiments*.

Grounding Language to Environments As discussed in the previous sections, specifically Sections 4.1.2 and 4.1.3, for the successful integration of LLMs or VLMs directly into robots, it is crucial to establish a connection between the language output and either skills or low-level actions. `SayCan` [24] learned a value function to score the joint likelihood between task-level plans and skills acquired through reinforcement learning or behavior cloning. `ZSP` [219] utilized semantic similarities between language instructions and skill names to ground robot skills, while `ProgPrompt` [221] and `Code as Policy` ground robot skills via program code synthesis. However, these skills are trained specifically for the environments in which they were tested. Therefore, we categorize all methods that associate language with skills as grounding to environments. Some research efforts, such as `Grounded Decoding` [233], attempt to address this by grounding foundation models to skills trained in a single environment, using small language-conditioned models trained in various environments as a probabilistic filter. Nonetheless, this approach is still limited as it presupposes the availability of grounded models suited for the environment in which it is embodied. In addition to skill grounding, work like `Voxposer` [227] have attempted to use LLMs to generate code for constructing value maps of the environment, which can then be planned on using existing planners. This represents a more general approach to integrating contextual information about the environment, compared to grounding to skill libraries.

Grounding Concepts to Environments The term 'concepts' refers to the unified latent representations, derived from training with diverse input data. Approaches such as `CLIP-Fields` [202], `VLMap` [201], and `NLMaps` [25] have endeavored to project `CLIP` visual representations and semantic label representations directly onto 3D point clouds. Beyond constructing explicit 3D maps, `GLAM` [234] proposes using reinforcement learning to ground language models with an internal map through interaction with environments. This approach demonstrated that LLMs can effectively function as Reinforcement Learning (RL) agents in textual environments. However, the challenge of generalizing these approaches to different environments and tasks, either by grounding concepts to point clouds or through implicit mapping, remains unresolved in current literature.

Grounding Language to Embodiments We categorize language grounding to embodiments as necessitating a specific condition: it must be agnostic to different environments. This is akin to having a universal interface that translates language into actions. Initiatives like `Prompt2Walk` [235] and `Saytap` [26] have experimented with directly using LLMs to generate joint angles or foot patterns in the language space for locomotion control.

Grounding Concepts to Embodiments Grounding concepts to embodiments involves directly anchoring foundation models to output joint torques, circumventing intermediary interfaces such as text. A notable example is `Gato` [30], which demonstrates versatility in tasks like playing Atari, captioning images, conversing, and manipulating objects with a robotic arm. `Gato` dynamically decides its output format—be it text, joint torques, button presses, or other tokens—based on the context of the task at hand. Another related development is `RT-2` [28], which, despite specifying the end-effector space in textual form, is capable of directly generating executable commands for robot manipulator operation.

In conclusion, the primary advantage of grounding language to environments and embodiments is the ease of implementing zero-shot learning without the need for additional training. However, significant drawbacks exist. For concepts not describable

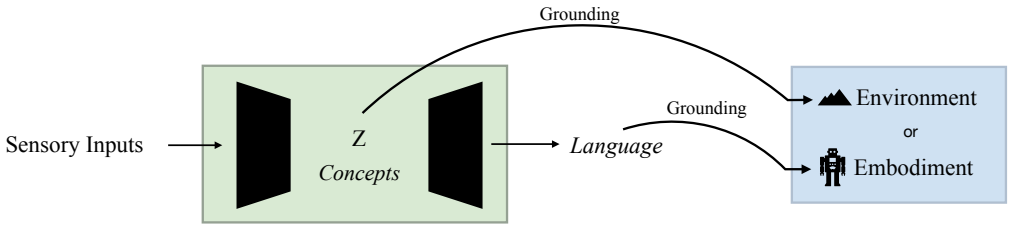


Figure 9: Grounding in robotics. We divide grounding techniques as grounding to environment and grounding to robotic embodiments.

by language, such as the nuances of finger movements, grounding to embodiments may fail. Moreover, reliance on a fixed set of skill libraries limits adaptability to diverse environments. Consequently, direct grounding from the latent concept space appears to be a more viable solution. Approaches that utilize interaction data [234] or expert data [28] have both shown promising results in addressing these challenges.

4.1.5 Data Generation with LLMs and VGMs

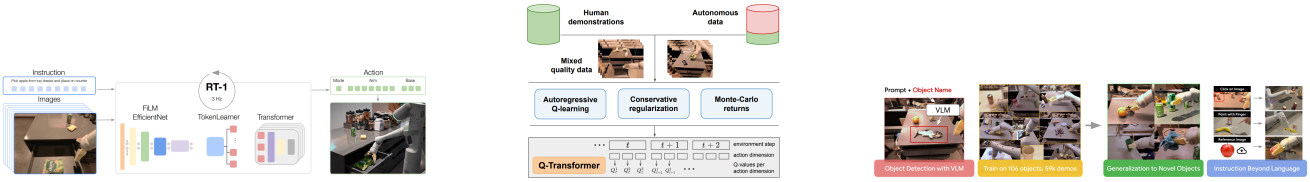
Recently, we have witnessed the power of content generation ability of LLMs and VGMs. Utilizing this ability, researchers have begun attempts to address the data scarcity problem by using foundation models to generate data. Ha *et al.* [236] propose a framework, '*scaling up and distilling down*', which, given natural language instructions, can automatically generate diverse robot trajectories labeled with success conditions and language. `RoboGen` by Wang *et al.* [237] further enhances this approach by incorporating automatic action trajectory proposals within a physics-realistic simulation environment, Genesis, enabling the generation of potentially-infinite data. Nevertheless, these approaches still face limitations: the data generated suffers from low diversity in assets and robot morphologies, issues that could be ameliorated with advanced simulations or hardware. `GenSim` [222] by Wang *et al.* proposes to generate novel long-horizon tasks with LLMs given language instructions, leading to over 100 simulation tasks for training language-conditioned multitask robotic policy. This framework demonstrates task-level generalization in both simulation and the real world, and sheds some light on how to distill foundation models to robotic policies through simulation programs. `ROSIE` by Yu *et al.* [238] uses a text-guided image generator to modify the robot's visual observation to perform data augmentation when training the control policy. The modification commands are from the user's language instruction, then the augmentation regions are localized by the open vocabulary segmentation model. `RT-Trajectory` [239] generates trajectories for the policy network to condition on. The trajectory generation also helps the task specification in the robot learning tasks. Black *et al.* [240] use a diffusion-based model to generate subgoals for a goal-conditioned RL policy for manipulation [241].

4.1.6 Enhancing Planning and Control Power through Prompting

The technique of Chain-of-Thought, as introduced by Wei *et al.* [242], compels the LLM to produce intermediate steps alongside the final output. This approach leverages a broader context window to list the planning steps explicitly, which enhances the LLM's ability to plan. The underlying reason for its effectiveness is the GPT series' nature as an autoregressive decoder. Semantic similarities are more pronounced between instructions to steps and steps to goal, than between instructions to the direct output. Nonetheless, the sequential nature of the Chain-of-Thought implies that a single incorrect step can lead to exponential divergence from the correct final answer [243].

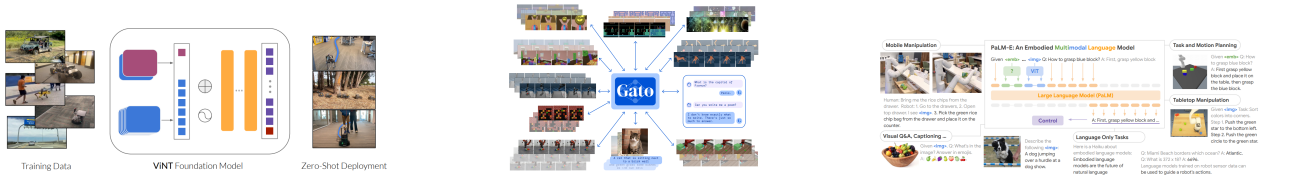
Alternative methodologies attempt to remedy this by explicitly listing steps within graph [244] or tree structures [245], which have demonstrated improved performance. Additionally, search-based methods such as Monte Carlo Tree Search (MCTS) [246] and Rapidly-exploring Random Tree (RRT) [225] have been explored to augment planning capabilities.

Furthermore, translating goal specifications from natural language into external planning languages, such as the Planning Domain Definition Language (PDDL), has also been shown to increase planning accuracy [247]. Finally, as opposed to an open-loop prompting style, iterative prompting approaches that incorporate feedback from the environment provide a more grounded and precise enhancement for long-term planning capability [31, 248].



(a) Imitation learning: RT-1 [27] (b) Reinforcement learning: Q-Transformer [195] (c) Vision language pretraining: MOO [249]

Robotic Foundation Models for: **manipulation**



(d) Motion planning model: VINT [29] (e) General purpose model: Gato [30] (f) General purpose model: PaLM-E [31]

Robotics Foundation Models for: **motion planning for navigation and multipurpose task**

Figure 10: Examples of Robotics Foundation Models (RFMs)

4.2 Robotics Foundation Models (RFMs)

With the increasing amount of robotics datasets, containing state-action pairs from real robots, the class of Robotics Foundation Models (RFMs) have likewise become increasingly viable [28, 29, 176]. These models are characterized by the use of robotics data to train them, in order to solve robotics tasks. In this subsection, we summarize and discuss different types of RFMs. We will first introduce RFMs that can perform one set of tasks, within one of the robotic modules from Section 2.1, which is defined as *single-purpose* Robotic Foundation Models. For example, an RFM that can generate low-level actions to control the robots, or a model that can generate higher-level motion plans. We later introduce RFMs that can carry out tasks in multiple robotic modules, hence *general-purpose* models that can perform perception, control, and even non-robotic tasks [30, 31].

4.2.1 Robotics Action Generation Foundation Models

Robotic action foundation models could take raw sensory observations, e.g., images or videos, and learn control output that is directly applied to robotic end-effectors. Models in this category include `RT` series [27, 28, 176], `RoboCat` [195], `MOO` [249], etc. According to the papers' results, these models show generalization in robot control tasks such as manipulation.

Imitation Learning Language inputs have also been used for providing task specifications for end-to-end direct downstream control of robotics, forming *language-conditioned* imitation policy learning. Li *et al.* [250] use pre-trained language models to initialize policy networks that predict the next action. The control policy is fine-tuned via behavior cloning and active learning, to improve task completion. Directly using language inputs to generate robot actions for improving human-robot collaboration has also been proposed: the Language-Informed Latent Actions (`LILA`) framework [251] learns to use language to modulate the low-level controller, effectively using language to map a 7-DoF robotic arm output to a 2-DoF input. Further, Hu *et al.* [252] use multi-agent reinforcement learning to fine-tune an LLM policy to enable humans to specify what kind of strategies they expect from their AI partners. Building on the theme of instructing robots through language, Lynch *et al.* [253] use behavioral cloning on a dataset of hundreds of thousands of language-annotated trajectories to improve vision-language-motor skills in the real world. Though previous works mostly take a language-conditioned approach, `RoboCat` [195] trains imitation learning policy with just robotic data.

Reinforcement Learning at Scale As discussed in Section 2.1.3 (Preliminaries), offline RL has the potential to learn good policies without even interacting with the environment. With the availability of large-scale robotic datasets, offline RL starts to play an important role in developing effective RFMs. Early large-scale offline RL models such as `QT-OPT` [112] use a Q-learning-based approach in an offline manner to learn policy from robotics data which are collected by a robot farm. The successors of `QT-OPT` extend it to multitask learning by incorporating multi-task curriculum or predictive information [254, 255]. Recently, with the success of Transformer models, Q-learning based on transformer (`Q-Transformer`) also shows

its potential [256]. PTR [257] is another promising work that adopts Conservative Q-Learning (CQL) [258] in a multi-task learning setting. We look forward to seeing more RL-based robotic foundation models.

Vision and Language Pre-training Another direction of action foundation models involves vision or language pre-training [28, 259–264]. For example, inspired by the great generalization ability of self-supervised vision-based pre-training, MVP by Radosavovic *et al.* [261] trains visual representations on real-world images and videos from the internet and egocentric video datasets, via a masked autoencoder, and demonstrate the effectiveness of scaling visual pre-training for robot learning. Following this work, RPT [262] proposes Mask-pretraining with real robot trajectory data. VC-1 [264] actually did a comprehensive study on the effectiveness of vision-based pre-training on policy learning. We also recommend readers to learn more information about this problem from that paper. Despite the effectiveness of these visual pretraining methods, Hansen *et al.* [263] reexamined some of these methods and discovered significant domain gaps and propose a learning from scratch approach which remains competitive. This provides us a new perspective to think about visual pretraining in robotics.

Beyond just using visual information, RT-2 [28] and Moo [249] use vision and language pre-trained model as a control policy backbone. PaLM-E [31] and PALI-X [265] were used to transfer knowledge from the web into robot actions. Slightly different from previous methods, VRB [266] learns affordance functions (instead of the policy itself) from large-scale video pertaining, providing another thought process for us to study how RFMs may generalize in real-world tasks.

Robotics Motion Planning Foundation Models Recently we have seen the rise of RFMs especially used for motion-planning purposes in visual navigation tasks [29, 267, 268]. These foundation models take advantage of the large-scale heterogeneous data and show generalization capability in predicting high-level motion-planning actions. These methods rely on rough topological maps [29, 267] consisting of only image observations instead of accurate metric maps and accurate localization as in conventional motion-planning methods (as described in Section 3.3). Unlike vision and language foundation models applied to motion planning, the robotic motion planning foundation model is still quite in its early stages.

4.2.2 General-purpose Robotics Foundation Models

Developing general-purpose robotic systems is always a holy grail in robotics and artificial intelligence. Some existing works [30, 31] take one step towards this goal. Gato [30] proposes a multimodal, multi-task, and multi-embodiment generalist foundation model that can play Atari games, caption images, chat, stack blocks with a real robot arm, and more—all with the same model weights. Similar to Gato, PaLM-E [31] is also a general-purpose multimodal foundation model for robotic reasoning and planning, vision-language tasks, and language-only tasks. Although not proven to solve all the robotics tasks that we introduced in Section 2, Gato and PaLM-E show a possibility of merging perception and planning into one single model. Moreover, Gato and PaLM-E show promising results of using the same model to solve various seemingly-unrelated tasks, highlighting the viability of general-purpose AI systems. Designed especially for robotic tasks, PACT [269] proposes one transformer-based foundation model with common pre-trained representations that can be used in various downstream robotic tasks, such as localization, mapping, and navigation. Although we have not seen many unified foundation models for robotics, we would expect more endeavors in this particular problem.

4.3 How Foundation Models Can Help Solve Robotics Challenges

In Section 3, we listed five major challenges in Robotics. In this section, we summarize how foundation models—either vision and language models or robotic foundation models—could help resolve these challenges, in a more organized manner.

All the foundation models related to visual information, such as VFMs, VLMs, and VGMs, are used in the perception modules in Robotics. LLMs, on the other hand, are more versatile and can be applied in both planning and control. We also list RFMs here, and these robotic foundation models are typically used in planning and action generation modules. We summarize how foundation models solve the aforementioned robotic challenges in Table 1. We notice from this table that all foundation models are good at generalization in tasks of various robotic modules. Also, LLMs are especially good at task-specification. RFMs, on the other hand, are good at dealing with the challenge of dynamics model since most RFMs are model-free approaches. For robot perception, the challenges in generalization ability and model are coupled, since, if the perception model already has very good generalization ability, there's no need to get more data for domain adaptation or additional fine-tuning. In addition, the call for solving the safety challenge is largely missing, and we will discuss the particular problem in Section 6.

Modules	Foundation Models	Generalization 3.1	Data 3.2	Model 3.3	Task Specification 3.4	Uncertainty 3.5
Perception 2.1.1	VFM 2.2.1	Conceptgraphs [270]	Conceptgraphs [270]	-	-	-
	VGM 2.2.2	-	ROSIE [238] RoboGen [237]	-	-	
	VLM 2.2.4	NLMap [25]	NLMap [25] RT-Traj. [239]	-	RT-Traj. [239]	-
Task Planning and Action Generation 2.1.2 and 2.1.3	LLM 2.2.3	SayCan [24]	SayCan [237] RT-Traj. [239]	RAP [271]	LILA [251] L2R [226]	KNOWNO [187]
	RFM 4.2	RT-1 [27] RT-2 [28] RoboCat [195] VINT [29]	RT-X [176]	RT-1 [27] RT-2 [28] RoboCat [195]	Zest [181]	-

Table 1: A summary of foundation models solving robotic challenges. Here we only list part of the works due to space limit. We find that uncertainly and safety are still largely unexplored.

Foundation Models for Generalization Zero-shot generalization is one of the most significant characteristics of current foundation models. Robotics benefits from the generalization ability of foundation models in nearly all aspects and modules. For the first one, generalization in perception, VLM and VFM are great choices as the default robotics perception models. The second aspect is the generalization ability in task-level planning, with details of task plans generated by LLMs [24]. The third one is in generalization in motion-planning and control, by utilizing the power of RFMs.

Foundation Models for Data Scarcity Foundation models are crucial in tackling data scarcity in robotics. They offer a robust basis for learning and adapting to new tasks with minimal specific data. For example, recent methods utilize foundation models to generate data to help with training robots, such as robot trajectories [236] and simulation [237]. These models excel in learning from a small set of examples, allowing robots to quickly adapt to new tasks using limited data. From this perspective, solving data scarcity is equivalent to solving the generalization ability problem in robotics. Apart from this aspect, foundation models—especially LLMs and VGMs—could generate datasets for robotics used in training perception modules [238] (see Section 4.1.5, above), and for task-specification [239].

Foundation Models to Relieve the Requirement of Models As discussed in Section 3.3, building or learning a model—either a map of the environment, a world model, or an environmental dynamics model—is vital for solving robotic problems, especially in motion-planning and control. However, the powerful few/zero-shot generalization ability that foundation models present may break that requirement. This includes using LLMs to generate task plans [24], using RFMs to learn model-free end-to-end control policies [27, 256], etc.

Foundation Models for Task-Specification Task-specifications as language prompts [24, 27, 28], goal images [181, 272], videos of humans demonstrating the task [273, 274], rewards [26, 182], rough scratch of trajectory [239], policy sketches [275], and hand-drawn images [276] have allowed goal specifications in a more natural, human-like format. Multimodal foundation models allow users to not only specify the goal but also help resolve ambiguities via dialogue. Recent work in understanding trust and intent recognition within the human-robot interaction domain has opened up newer paradigms in our understanding of how humans use explicit and implicit cues to convey task-specifications. While significant progress has been made, recent work in prompt engineering for LLMs implies that even with a single modality, it is challenging to generate relevant outputs. Vision-Language Models are proven to be especially good at task-specification, showing potential for resolving this problem in robotics. Extending the idea of vision-language-based task-specifications, Cui *et al.* [181] explore methods to achieve multi-modal task specification using more natural inputs like images obtained from the internet. Brohan *et al.* [27] explores this idea of zero-shot transfer from task-agnostic data further, by providing a novel model class that exhibits promising scalable model properties. The model encodes high-dimensional inputs and outputs, including camera images, instructions, and motor commands into compact token representations to enable real-time control of mobile manipulators.

Foundation Models for Uncertainty and Safety Though being a critical problem in robotics, uncertainty and safety using foundation models for robotics is still underexplored. Existing works like KNOWNO [187] proposes a framework for measuring and aligning the uncertainty of LLM-based task planners. Recent advances in Chain-of-Thought prompting [277], open vocabulary learning [278], and hallucination recognition in LLMs [279] may open up newer avenues to address these challenges.

5 Review of Current Experiments and Evaluations

In this section, we summarize the datasets, benchmarks, and experiments of the current research works.

5.1 Datasets and Benchmarks

Relying solely on knowledge learned from language and vision datasets is limiting. Some concepts, like friction or weight, are not easily learned through these modalities alone, as suggested by Gao *et al.* [280] and Tatiya *et al.* [200] in their works on physically grounded VLMs. Therefore, in order to make robotic agents that can better understand the world, researchers are not just adapting foundational models from the language and vision domains; they are also advancing the development of large, diverse, and multimodal *robotic datasets* for training or fine-tuning these foundation models. This effort is now diverging into two directions: collecting data from the real world, versus collecting data from simulations and then transferring it to the real world. Each direction has its pros and cons. We will cover these datasets and simulations in the following paragraphs and discuss their respective advantages and disadvantages.

5.1.1 Real World Robotics Datasets

Real-world robotics datasets are highly appealing due to their diverse object classes and multimodal inputs, offering a rich resource for training robotic systems without the need for complex and often inaccurate physical simulations. However, creating these large-scale datasets presents a significant challenge, primarily due to the absence of a substantial 'data flywheel' effect. This effect, which greatly benefited fields like CV and NLP through contributions from millions of internet users, is less evident in robotics. The limited incentive for individuals to upload extensive sensory inputs and corresponding action sequences poses a major hurdle in data acquisition. Despite these challenges, current efforts are focused on addressing these gaps. RoboNet [281] is a notable effort in this direction, offering a large-scale, diverse dataset across different robotic platforms for multi-robot learning. Bridge Dataset V1 [282] collects 7200 hours of demonstrations in real household kitchen manipulation tasks, and its following Bridge-V2 [283] contains 60,096 trajectories collected across 24 environments on common low-cost robots. Language-Table [253] collects 600,000 language-labeled trajectories—an order of magnitude larger than prior available datasets. RT-1 [27] contains 130k episodes that cover 700+ tasks, collected using a fleet of 13 Google mobile manipulation robots, over 17 months. While the aforementioned datasets represent significant advancement over prior lab-scale datasets, offering a relatively large volume of data, they are limited to single modalities or specific robot tasks.

To overcome these limitations, some recent initiatives have made notable progress. For example, GNM [267] successfully integrated six different large-scale navigation datasets, utilizing a unified navigation interface based on waypoints. Furthermore, a recent collaborative effort among various laboratories called RT-X [176] has aimed to standardize data across different datasets, by using a 7-degree-of-freedom end-effector's pose as a universal reference across different embodiments. This approach facilitates the joint use of diverse datasets and showed positive performance in cross-morphology transfer learning.

Building on these advancements, the scale of real-world robotics datasets is beginning to grow, albeit still lagging behind the immense volume of internet-scale language and vision corpora. The accessibility of advanced hardware such as the Hello Stretch Robot, Unitree Quadrupeds, and open-source dexterous manipulators [284] is expected to catalyze this growth. As these technologies become more widely available, they are likely to initiate the desired 'data flywheel' effect in Robotics.

5.1.2 Robotics Simulators

While we await the widespread deployment of robotic hardware to gather massive amounts of robot data, another approach is to develop simulators that closely mimic real-world graphics and physics. The advantage of using simulation is the ability to deploy tens of thousands of robot instances in a simulated world, enabling simultaneous data collection.

Simulators focus on different aspects, such as photorealism, physical realism, and human-in-the-loop interactions. For navigation tasks, photorealistic simulators are crucial. AI Habitat addresses this by utilizing realistically-scanned 3D scenes from the Matterport3D [285] and Gibson [286] datasets. Furthermore, Habitat [287] is a simulator that allows AI agents to navigate through various realistic 3D spaces and perform tasks, including object manipulation. It features multiple sensors and handles generic 3D datasets. Habitat 2.0 [174] builds upon the original by introducing dynamic scene modeling, rigid-body physics, and increased speed. Habitat 3.0 [175] further integrates programmable humanoids to enhance the simulation

experience. Additionally, the AI2THOR simulator [288] is another promising framework for photorealistic visual foundation model research, as evidenced in [201, 289]. Other simulators, like Mujoco [170], focus on creating physically realistic environments for advanced manipulation and locomotion tasks.

Moreover, simulators like AirSim [290] and the Arrival Autonomous Racing Simulator [102], both built on Unreal Engine, provide a balance of reasonable physics and photorealism. Ultimately, while the aforementioned simulators excel in various areas, they face common challenges such as parallelism. Simulators like Issac Gym [171] and Mujoco 3.0 [291] have attempted to overcome these challenges by using GPU acceleration to expedite the data-collection process.

Despite the abundance of data available in simulators, there are inherent challenges in their use. Firstly, the domain gap between simulations and the real world makes transferring from sim to real problematic—issues that early works are already seeking to resolve [164]. Secondly, the diversity of environments and base objects is still lacking. Therefore, to effectively utilize simulations in the future, continuous improvements in these two areas are essential.

5.2 Analysis of Current Method Evaluation

We conduct a meta-analysis of the experiments of papers listed in Tables 2 to 7 and Figure 11, encouraging readers to consider the following questions

1. What tasks are being solved?
2. On what datasets or simulators have they been trained? What robot platforms are used for testing?
3. What foundation models are being utilized? How effectively are the tasks solved?
4. What base foundation models are more frequently used in these methods?

We summarize several key trends observed in the current literature concerning the experiments conducted:

Imbalanced Focus among Manipulation Tasks: There is a significant emphasis on general pick-place tasks, particularly tabletop and mobile manipulation. This is likely due to the ease of training for tabletop gripper-based manipulation skills and their potential to form skill libraries that interact with foundation models. However, there is a lack of extensive exploration in low-level action outputs, such as dexterous manipulation and locomotion.

Need for Improved Generalization and Robustness: Generalization and robustness of end-to-end foundational robotics models have room for improvement. In tabletop manipulation, the use of foundation models leads to performance drops ranging from 21% [27, 227] to 31% [28] in unseen tasks. In addition, these models still need improved robust to disturbances, performance drops 14% [27] to 18% [227] for similar tasks.

Limited Exploration in Low-Level Actions: There remains a gap in the exploration of direct low-level action outputs. The majority of research focuses on task-level planning and utilizes foundation models with pre-trained or pre-programmed skill libraries. However, existing papers [28, 30, 176] that explore low-level action outputs mainly focus on table-top manipulation, where the action space is limited to the end effector's 7 degrees of freedom (DoF). Models that directly output joint angles for tasks like dexterous manipulation and locomotion still require a more thorough research cycle.

Control Frequencies Too Slow to be Deployed on Real Robots: Most current approaches to robotic control are open-loop, and even those that are closed-loop face limitations in inference speed. These speeds typically range from 1 to 10 Hz, which is considered low for the majority of robotics tasks. Particularly for tasks like humanoid locomotion, a high-frequency control of around 500 Hz is required for the stabilization of the robot's body [322].

Lack of Uniform Benchmarks for Testing: The diverse nature of simulations, embodiments, and tasks in robotics leads to varied benchmarks, complicating the comparison of results. Additionally, while success rate is often used as the primary metric, it may not sufficiently evaluate the performance of real-world tasks involving large foundation models, as latency is not captured by the success rate alone. More nuanced evaluation metrics that consider inference time, such as the Compute Aware Success Rate (CASR) [225].

Table 2: Tabletop Manipulation

Title	Datasets & Simulation	Real Robot	Number of Tasks	Base Model	SR	SR Descriptions	Frequency
RT-X	Kitchen Manipulation [292], Cable Routing [293], NYU Door Opening [294], AUTOLab UR5 [295], Robot Play [296], Bridge [283], RL Bench (RT-1) [27]	Google robot	160266	RT-2(PaLI)	30% 63% 92%	Bridge, Small lab datasets, Google robot	3-10Hz
RT-2	RL Bench (RT-1) [27] Language-Table [253]	Google robot	480+	PaLI, PaLM-E	93% 62%	Seen tasks, Unseen tasks	1-5Hz
RT-1[27]	RL Bench (RT-1) [27]	Google robot	744	RT-1	97% 76% 83% 59%	Seen Tasks, Unseen Tasks, With Distractors, Novel background	3 Hz
GNFactor [198]	RL Bench (RT-1) [27]	XArm7	166	Stable Diffusion, CLIP	32% 28%	Multi-task test, Unseen tasks	Unspec.
MOO [249]	Self-created robotic data	Google robot	106	Owl-ViT, RT-1	92% 75%	Seen Tasks, Unseen Tasks	Unspec.
PhysObjects [280]	PhysObjects [280]	Franka Panda	51	FlanT5-XXL, GPT-4	88% 88%	PhysObjects Test Set, Real scenes	open loop
Matcha [297]	CoppeliaSim [298]	NiCOL	50	ViLD, text-davinci-003	91% 57%	Distinct descriptions, Indistinct descriptions	open loop
Scalingup [236]	Scalingup benchmark [236]	UR5 arm	18	GPT-3	79%	Mean Success	35 Hz
F3RM [197]	PyBullet [299] and Self-created robotic data	Franka Panda	18	CLIP	78% 62%	Grasp-Place, Object Manip.	Unspec.
VIMA [272]	VIMABench [272]	None	17	pre-trained T5 tokenizer	81% 81% 79% 49%	L1, L2, L3, L4	Unspec.
Instruct2Act [300]	VIMABench [272]	None	17	CLIP, SAM, text-davinci-003, LLaMA	84%	Mean Success	open loop
VoxPoser [227]	Saipen [301] Where2act [302]	Franka Emika	13	GPT-4, SAM, Owl-ViT	88% 70% 67%	Static environments, With disturbances, Unseen semarios	5HZ
Text2Motion [220]	TableEnv [220]	Franka Panda	6	text-davinci-003	82%	Mean Success	open loop
GenSim [222]	PyBullet [299] and Self-created robotic data	XArm7	100+	GPT-4	53.3% 68.8%	50 Tasks 70 Tasks	Unspec.

Table 3: Dexterous Manipulation

Title	Datasets & Simulation	Real Robot	Number of Tasks	Base Model	SR	SR Descriptions	Frequency
RoboCat [195]	RGB-Stacking Benchmark [303]	Sawyer 7-DoF, Panda 7-DoF, KUKA 14-DoF	253	RoboCat	82%, 74%, 86%	Sawyer in sim, Panda in sim, KUKA in real	10 Hz 20 Hz

Table 4: Mobile Manipulation

Title	Datasets & Simulation	Real Robot	Number of Tasks	Base Model	SR	SR Descriptions	Frequency
LLaRP [304]	Language Rearrangement [304]	None	1000	LLaMA-7B	42% 0% 8% 39%	Total, Multiple objects, Spatial Reasoning, Conditional instuct.	open loop
Code-As-Policies [216]	Customized RoboCodeGen [216], HumanEval [305]	Eveyday Robots, UR5 Robot arm	214	Codex, GPT-3	95% 96%	RoboCodeGen, HumanEval	open loop
InnerMonlogue [248]	Ravens [306];	UR5e robot, Google robot	130	ViLD GPT-3.5	51% 50% 90% 60%	Seen Ravens tasks, Unseen Ravens tasks, Real robot arm, Google robot	Unspec.
LIV [307]	MetaWorld [308] FrankaKitchen [260]	Franka panda	114	LIV	30% 30% 45%	FrankaKitchen, MetaWorld, Real robots	15hz
SayCan [24]	proprietary simulator and self-created robotic data	Google robot	101	PaLM, Flan	74%	Mean Success	open loop
PaLM-E [31]	Lang-table [253]	Google robot, xArm6	100	PaLM-E	83% 76% 52% 91%	Grasping, Stacking, Lang-table tasks, Google robot	1-5Hz
TidyBot [309]	TidyBot [309]	TidyBot	96	ViLD, CLIP , text-davinci003	91% 83%	Sim, Real	open loop
LLM-Grop [310]	Gazebo [311]	Segway base + UR5e arm	8	GPT-3	4.08	Average human rating(1-5)	open loop
LLM+P [247]	International Planning Competition [223]	None	7	GPT-4	20% 90% 0% 95% 85% 20% 10%	Barman, BLOCKSWORLD, FLOORTILE, GRIPPERS, STORAGE, TERMES, TYREWORLD	open loop
HomeRobot [205]	Habitat [312]	Hello-Robot Stretch	8	-	20%	RL baseline	closed loop

Table 5: Navigation

Title	Datasets & Simulation	Real Robot	Number of Tasks	Base Model	SR	SR Descriptions	Frequency
LM-Nav [204]	Self-created datasets	Clearpath Jackal UGV	20	GPT-3	80%	Mean Success	open loop
CLIP-Fields [202]	Habitat-Matterport 3D Semantic [312]	Stretch Robot	14	CLIP Detic	79%	Mean Success	open loop
GNM [267]	GNM datasets [267]	LoCoBot Vizbot DJI Tello Jackal UGV	3	GNM	96%	Mean Success	Unspec.

Table 6: locomotion

Title	Datasets & Simulation	Real Robot	Number of Tasks	Base Model	SR	SR Descriptions	Frequency
SayTap [26]	IsaacGym [171]	Unitree A1	30	GPT-4	97%	Mean Success	openloop
Prompt2Walk [229]	Mujoco [170]	None	1	GPT-4	80%	Mean Success	10 Hz

Table 7: Multi-Tasks

Title	Datasets & Simulation	Real Robot	Number of Tasks	Base Model	SR	SR Descriptions	Frequency
Gato [30]	Meta-World [308], Sokoban [313], BabyAI [314], Procgen Benchmark [315], Arcade Learning Environment [316], DM Control Suite [317]	Sawyer arm	604	GPT-4	97% 68% 80% 68% super-human 64%	Meta-World, Sokoban, BabyAI, Procgen, ALE Atari, DM Control Suite	20 Hz
Grounded Decoding [318]	2D Maze [319], Ravens [306]	None	124	GPT-3.5 PaLM	71% 46% 95% 51%	Ravens Seen Tasks, Ravens Seen Tasks, 2D Maze, Mobile Manipulation	open loop
Eureka [182]	Issac Gym [171], Bidexterous Manipulation [320]	None	29	GPT-4	55% 3.7	Bi-Dextrous, Isaac Tasks – (Human Score)	open loop
Lang2Reward [226]	MuJoCo MPC [321]	Google robot	17	GPT-4	95% 80%	Quadruped, Dextrous manipulator	open loop
VC-1 [264]	CortexBench [264]	None	17	VC-1	59% 89% 67% 72% 60% 70% 63%	Adroit, Meta-World, DMControl, Trifinger, ObjectNav, ImageNav, Mobile Pick	open loop

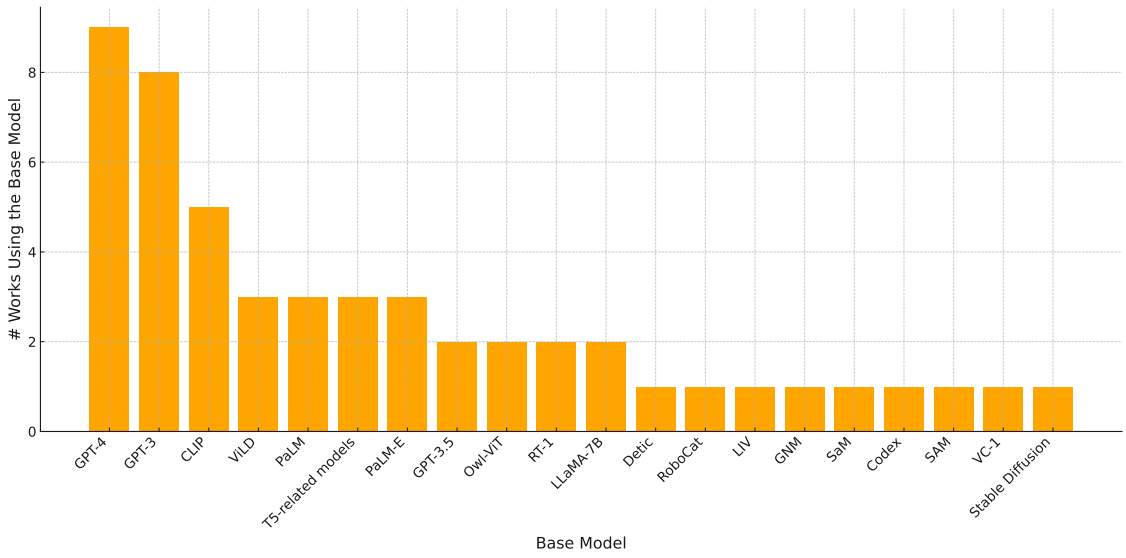


Figure 11: The histogram showing the number of times different base foundation models are used in developing robotics systems, among the papers we included in this survey. In the plot we can see GPT-4, GPT-3 are among top choices due to their few-shot promptable nature, as well as accessibility through APIs. CLIP and ViLD are frequently used to bridge image and text representations. Apart from CLIP, T5 family models are frequently used to encode text to get text features. PaLM and PaLM-E are used for robot planning. RT-1, which is originally developed for manipulation, emerges as a new base model which other manipulation models are built upon.

6 Discussions and Future Directions

6.1 Remaining Challenges and Open Discussions

Grounding for Robot Embodiment Although numerous strategies have been explored to address the problem of grounding, as discussed in Section 9, there are many open challenges in this area. First, grounding needs an effective medium or interface that bridges concepts and robot actions. Existing interfaces, such as those employing natural language [24, 31] and code [216, 221], are limited. While concepts can be articulated through language and code, they are not universally applicable to nuances such as dexterous body movements. Furthermore, these interfaces often depend on predefined skill libraries that are not only time-intensive to develop but also lack generalization to new environments. Using reward as an interface [182, 226, 232] may alleviate some of the generalization issues in simulations by acquiring skills dynamically. However, the time-consuming and potentially unsafe nature of training RL algorithms in the real world raises questions about the feasibility of this method, with real-world validations of its effectiveness yet to be demonstrated.

Second, we need to move from a unimodal notion of grounding, like mapping the word to meaning to a more holistic grounding of multiple sensory modalities. Approaches that rely solely on visual data [280] may capture certain physical properties such as material, transparency, and deformability. Yet, they fall short in grasping concepts like friction, which requires interactive data with proprioceptive feedback, or the scent of an object, which cannot be acquired without additional modalities such as olfaction.

Lastly, we should consider grounding from an embodiment perspective. The same task may necessitate distinct actions based on the robot's embodiment; for example, opening a door would require drastically different maneuvers from a humanoid robot compared to a quadruped. Current research on grounding often emphasizes environmental adaptation while affording less consideration to how embodiment shapes interaction strategies.

Safety and Uncertainty As we pursue deployments of real robots to work alongside humans in factories, to provide elderly care, or to assist in homes and offices, these autonomous systems (and the foundation models that power them) will require more effective measures of safety. While formal hardware and software safety checks still apply, the use of foundation models to support provable safety analysis will become an increasingly necessary direction. With the goal of deploying robots to safety-critical scenarios, prior works have considered leveraging Lyapunov-style safety index functions [122, 323, 324], in

attempts to provide hard safety guarantees for nonlinear systems with complex dynamics and external disturbances (see also Section 3.5). Traditionally, the systems under consideration by the provable safety literature are of low dimension, often require careful specification of a world/dynamics model, require specifying an initial safe set and/or set-boundary distance functions, require some heuristics and training "tricks" to obtain useful safety value functions that balance conservativeness versus performance, do not naturally support multi-agent settings, and present challenges in *safely* updating the safety value function and growing the safe set online. Herbert *et al.* [324] synthesized several techniques into a framework—thereby easing computation, streamlining updates to the safe sets by one or more orders of magnitude compared to the prior art, and extending Hamilton-Jacobi Reachability analysis to 10-dimensional systems that govern quadcopter control. Chen *et al.* [122] combine RL with HJ Reachability analysis to learn safety value functions from high-dimensional inputs (RGB images, plus vehicle state), to trade-off a performance-oriented policy and a safety-oriented policy, within a jointly-optimized dual actor-critic framework, for simulated autonomous racing. Tian *et al.* [325] integrate HJ Reachability analysis in the context of multi-agent interactions in urban autonomous driving, by formulating the problem as a general-sum Stackelberg game.

However, in all of these works, open questions remain on integrating socially-acceptable safety constraints and formal guarantees for systems with robotic foundational models. One of the directions is to formulate safety as an affordance [326]. The definition of safety changes based on the capability of the robot and social context. Another focus for safety is to ensure robust alignment of the robot's inferred task specification to a human user's communicative intent. Foundation models offer a way to encode the enormous world knowledge, which can serve as commonsense priors to decode the underlying intent. Recent works improve the use of LLMs for robotics with conformal prediction [187] and explicit constraint checking [327]. Despite these advances, foundation models currently lack native capacity to reason about the uncertainty associated with their outputs. If properly calibrated, uncertainty quantification in foundation models can be used to trigger fall-back safety measures like early termination, pre-defined safe maneuvers, or human-in-the-loop interventions.

Is there a Dichotomy between End-to-End and Modular Approaches? The human brain serves as an example of a functional approach to learning and generalization. While neuroscientists have identified specific regions of the brain, such as the visual cortex, somatosensory cortex, and motor cortex, the brain demonstrates remarkable plasticity and the ability to reorganize its functions to adapt to changes or brain lesions. This flexibility suggests that the brain may have evolved to be modular as a consequence of unified training, combining specific functionalities while maintaining the capacity for general learning [328, 329]. Similarly, in "Bertology", NLP researchers show how local parts of trained networks can specialize in one area over others. This indicates that certain modules of large-scale models may become highly specialized for specific functions, which can be adapted for downstream tasks without re-training the entire network. This transfer learning approach can lead to more efficient use of computational resources and faster adaptation to new tasks.

In the context of robotics, taking a premature stand for either modular or end-to-end policy architectures may limit the potential of foundation models for robotics. Modular solutions can provide specific biases and enable effective task-specific performance, but they may not fully leverage the potential of general learning and transferability. On the other hand, end-to-end solutions have a history of working well on certain tasks on CV and NLP, but they might not offer the desired flexibility for adaptation to new situations. As [45] pointed out, there appears to be a misconception about the modular versus end-to-end dichotomy. This is because the former pertains to architecture while the latter relates to optimization – they are not mutually exclusive.

Regarding the architecture and optimization design for foudnation models used in robotics, we can focus on a functional approach rather than categorizing it as either modular or end-to-end differentiable. One of the goals of a robotic foundational model is to allow flexible modular components, each responsible for specific functionalities, with unified learning that leverages shared representations and general learning capabilities.

Adaptability to Physical Changes in Embodiment From employing a pen to flip a light switch to maneuvering down a staircase with a broken leg encased in a cast, the human brain demonstrates versatile and adaptable reasoning. It is a single unit that controls perceptual understanding, motion control, and dialogue capabilities. For motion control, it adapts to the changes in the embodiment, due to tool use or injury. This adaptability extends to more profound transformations, such as individuals learning to paint with their feet or mastering musical instruments with specialized prosthetics. We want to build such interactive and adaptable intelligence in Robotics.

In the previous discussions, we saw existing works successfully deploying navigation foundation models for various robot

platforms [29], such as different wheeled robots and quadrupedal robots. We also witnessed the manipulation foundation model used in different manipulators [28, 330] which can be used across different robotic platforms, ranging from tabletop robot arms to mobile manipulators.

one of the key open research question is how robotics foundational models should enable motion control across different physical embodiments. Robot policies deployed in homes and offices must be robust to mechanical motion failures, such as sensor malfunctions or actuator breakdowns, ensuring continued functionality in challenging environments. Furthermore, robotic systems must be designed to adapt to a variety of tools and peripherals, mirroring the human capability to interact with different instruments for specific tasks and physical tool uses. While some works [331–333] have explored learning representations for diverse tool use, these approaches are yet to be scaled up with foundation models.

World Model, or Model-agnostic? In classical robotics, especially in planning and control problems, it was common to attempt to model as much as possible about the world that would be needed for robotics tasks. This was often carried out by leveraging structural priors about the tasks, or by relying on heuristics or simplifying assumptions. Certainly, if it was possible to perfectly model the world, solving robotics problems would become a lot simpler. Unfortunately, due to the complexity of the real world, world modeling in Robotics remains extremely difficult and sometimes even intractable. As a consequence, obtaining policies that generalize across tasks and environments remains a core problem.

The foundation models surveyed in this paper mostly take a model-agnostic (model-free) approach, leveraging the strength of expansive datasets and large-scale deep learning architectures. Some exceptions have attempted to emulate model-based approaches by directly employing LLMs as dynamic models. However, these attempts are still constrained by the inherent limitations of text-only descriptions and are prone to encountering issues with hallucinations, as discussed in [225, 271]. Many researchers would argue [138] that the data-scaled learning paradigm of these foundation models is still quite different from how humanity and animals learn, which is in an extreme data- and energy-efficient manner. Achieving even close to the joint performance and efficiency of human learning ability remains intriguing. In [138], LeCun argues that one possible answer to resolving that puzzle may lie in the *learning* of world models, a model that predicts how the state of the world going to change as consequences of the actions taken. If we were to develop world models that can emulate the precision of the world's representation through rigorous mathematical and physical modeling, it would bring us significantly closer to addressing and generalizing complex issues in robotics. These sophisticated and reliable world models would enable the application of established model-based methodologies, including search-based and sample-based planning, as well as trajectory optimization techniques. This approach would not only facilitate the resolution of planning and control challenges in robotics but also augment the explainability of these processes. It is posited that the pursuit of a 'foundation world model', characterized by remarkable generalization abilities and zero-shot learning capabilities, holds the potential to be a paradigm-shifting development in the field.

Novel Robotics Platforms and Multi-sensory Information As demonstrated in Figure 4c and the Meta-analysis in Tables 2-7, existing real robot platforms utilized for deploying foundation models are predominantly limited to gripper-based, single-arm robot manipulators. The range of concepts learnable from tasks executed by these hardware systems is restricted, primarily because the simple opening and closing actions of a gripper are easily describable by language. To enable robots to achieve a level of dexterity and motor skills comparable to those of animals and humans, or to perform complex domestic tasks, it is essential for foundation models to acquire a deeper understanding of physical and household concepts. This learning necessitates a broader spectrum of information sources, such as diverse sensors (including smell, tactile, and thermal sensors), and more intricate data such as proprioception data from robot platforms with high degrees of freedom.

Current dexterous manipulators, e.g., Shadow Hand [334], are prohibitively expensive and prone to frequent breakdowns, hence they are predominantly experimented with in simulation. Moreover, tactile sensors are still limited in their application, often confined to the fingertips, as in [335], or offer only low resolution, as observed in the robot-sweater [336]. Furthermore, since the bulk of data-collection is still conducted through human demonstrations, platforms that facilitate more accurate and efficient data acquisition, such as ALOHA [337] and Leap Hands [284], are gaining popularity. Therefore, we posit that significant contributions are yet to be made—not only in terms of software innovations, but also in hardware. These advancements are crucial for providing richer data-collection and, thus, expanding the conceptual space of robotics foundation models.

Continual Learning Continual learning broadly refers to the ability to learn and adapt to dynamic and changing environments. Specifically, it refers to learning algorithms that can learn and adapt to the underlying training data distribution and changing learning objective, as they evolve through time.

Continual learning is challenging, as neural network models often suffer from catastrophic forgetting, leading to a significant decrease in overall model performance on prior tasks. One naive solution to mitigate performance degradation due to catastrophic forgetting involves periodically re-training models with the entire dataset collected, which generally allows models to avoid forgetting issues, since the process encompasses both old and new data. However, this method demands significant computational and memory resources. In contrast, training or fine-tuning only on new tasks or current data, without revisiting previous data, is less resource-intensive but incurs catastrophic forgetting due to the model's tendency to overwrite previously learned information. This forgetting can be attributed to task interference between old and new data, concept drifts as data distributions evolve over time, and limitations in model expressivity based on their size.

Additionally, with the increasing capacities of models, continuously re-training them on expanding data corpora becomes less feasible. Recent works in vision and language continual learning [338–341] have proposed various solutions, yet achieving effective continual learning, that can be applied to robotics, still remains a challenging objective. For continual learning, large pre-trained foundational models currently face the above challenges and more, primarily because their extensive size makes retraining more difficult. In Robotics applications, specifically, continual learning is imperative to the deployability of robot learning policies in diverse environments, yet it is still a largely-unexplored domain. Whereas some recent works have studied various sub-topics of continual learning [342]—e.g., incremental learning [343], rapid motor adaptation [344], human-in-the-loop learning [345, 346]—these solutions are often designed for a single task/platform and do not yet consider foundation models.

We need continual learning algorithms that are designed with machine learning fundamentals in mind and practical real-time systems considerations. Some open research problems and viable approaches are: (1) mixing different proportions of the prior data distribution when fine-tuning on latest data to alleviate catastrophic forgetting [347], (2) developing efficient prototypes from prior distributions or curriculum to learn new tasks [348] for task inference, (3) improving training stability and sample-efficiency of online learning algorithms [349, 350], and (4) identifying principled ways to seamlessly incorporate large-capacity models into control frameworks (perhaps with hierarchical learning [351–353] / slow-fast control [354]) for real-time inference.

Standardization and Reproducibility The robotics community needs to encourage standardized and reproducible research practices to ensure that published findings can be validated and compared by others. To enable reproducibility at scale, we need to bridge the gap between simulated environments and real-world hardware and improve the transferability of ML models. Homerobot [205] is a promising step towards enabling both simulation and hardware platforms for open vocabulary pick-and-place tasks. We need to establish standardized task definitions and affordances to handle different robot morphologies, enabling more efficient model development.

6.2 Summary

In this survey paper, we analyzed the current research works on foundation models for robotics based on two major categories: (1) works which apply foundation models to robotic tasks, and (2) works attempting to develop robotics foundation models for robotics tasks using robotics data. We went through the methodologies and experiments of these papers, and provided analysis and insights based on these research works. Furthermore, we specially covered how these foundation models help resolve the common challenges in robotics. Finally, we discussed remaining challenges in robotics that have not been solved by foundation models, as well as other promising research directions.

Disclaimer

Due to the rapidly changing nature of the field, we checkpointed this version of literature review on Dec 13th 2023, and might have missed some relevant work. In addition, due to the rich body of literature and the extensiveness of this survey, there may be inaccuracies or mistakes in the paper. We welcome readers to send pull requests to our GitHub repository (inside **https://robotics-fm-survey.github.io/**) so we may continue to update our references, correct the mistakes and inaccuracies, as well as updating the entries of the meta studies in the paper. Please refer to the contribution guide in the GitHub repository.

Acknowledgments

We would like to thank Vincent Vanhoucke for feedbacks on a draft of this survey paper. In addition, we would like to thank Kedi Xu for insightful discussions about the papers list.

References

[1] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013. 3, 9

[2] Daniel Maturana, Po-Wei Chou, Masashi Uenoyama, and Sebastian Scherer. Real-time semantic mapping for autonomous off-road navigation. In *Field and Service Robotics*, pages 335–350. Springer, 2018. 9

[3] Berk Calli, Arjun Singh, James Bruce, Aaron Walsman, Kurt Konolige, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. Yale-cmu-berkeley dataset for robotic manipulation research. In *International Journal of Robotics Research*, page 261 – 268, 2017. 3

[4] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014. 3

[5] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017. 3

[6] William Shen, Felipe Trevizan, and Sylvie Thiébaux. Learning domain-independent planning heuristics with hypergraph networks. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 30, pages 574–584, 2020. 3

[7] Beomjoon Kim and Luke Shimanuki. Learning value functions with relational state representations for guiding task-and-motion planning. In *Conference on Robot Learning*, pages 955–968. PMLR, 2020.

[8] Grady Williams, Paul Drews, Brian Goldfain, James M. Rehg, and Evangelos A. Theodorou. Aggressive driving with model predictive path integral control. In *ICRA*, 2016. 3, 10

[9] Ahmed H Qureshi, Yinglong Miao, Anthony Simeonov, and Michael C Yip. Motion planning networks: Bridging the gap between learning-based and classical motion planners. *IEEE Transactions on Robotics*, pages 1–9, 2020. 3, 6

[10] Adam Fishman, Adithyavairavan Murali, Clemens Eppner, Bryan Peele, Byron Boots, and Dieter Fox. Motion policy networks. In *Proceedings of the 6th Conference on Robot Learning (CoRL)*, 2022. 3, 6

[11] Xue Bin Peng, Erwin Coumans, Tingnan Zhang, Tsang-Wei Lee, Jie Tan, and Sergey Levine. Learning agile robotic locomotion skills by imitating animals. In *RSS*, 2020. 3, 6

[12] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. In *Journal of Machine Learning Research*, 2016. 5, 6

[13] Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. In *Science Robotics*, 30 Jan 2019. 6

[14] Anusha Nagabandi, Kurt Konoglie, Sergey Levine, and Vikash Kumar. Deep dynamics models for learning dexterous manipulation. In *CoRL*, 2019. 3, 6

[15] Dmitry Kalashnkov and Jake Varley and Yevgen Chebotar and Ben Swanson and Rico Jonschkowski and Chelsea Finn and Sergey Levine and Karol Hausman. Mt-opt: Continuous multi-task robotic reinforcement learning at scale. *arXiv:2104.08212*, 2021. 3

[16] Eric Jang, Alex Irpan, Mohi Khansari, Daniel Kappler, Frederik Ebert, Corey Lynch, Sergey Levine, and Chelsea Finn. BC-Z: Zero-Shot Task Generalization with Robotic Imitation Learning. In *5th Annual Conference on Robot Learning*, 2021. 3

[17] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. 3, 8

[18] Aditya Ramesh, Prafulla Dhariwal Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 3, 7

[19] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar, Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 3, 7

[20] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021. 3, 7

[21] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 7

[22] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 3, 5, 7

[23] Rishi Bommasani et. al. from the Center for Research on Foundation Models (CRFM) at the Stanford Institute for Human-Centered Artificial Intelligence (HAI). On the opportunities and risks of foundation models. In *arXiv:2108.07258*, 2021. 3, 7

[24] Ahn et. al. Do as i can, not as i say: Grounding language in robotic affordances. In *CoRL*, 2022. 3, 11, 14, 16, 20, 24, 26

[25] Boyuan Chen, Fei Xia, Brian Ichter, Kanishka Rao, Keerthana Gopalakrishnan, Michael S. Ryoo, Austin Stone, and Daniel Kappler. Open-vocabulary queryable scene representations for real world planning. In *arXiv:2209.09874*, 2022. 12, 13, 16, 20

[26] Yujin Tang, Wenhao Yu, Jie Tan, Heiga Zen, Aleksandra Faust, and Tatsuya Harada. Saytap: Language to quadrupedal locomotion, 2023. 3, 11, 15, 16, 20, 25

[27] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022. 3, 9, 11, 12, 18, 20, 21, 22, 23

[28] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alex Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *arXiv preprint arXiv:2307.15818*, 2023. 3, 11, 16, 17, 18, 19, 20, 22, 28

[29] Dhruv Shah, Ajay Sridhar, Nitish Dashora, Kyle Stachowicz, Kevin Black, Noriaki Hirose, and Sergey Levine. Vint: A foundation model for visual navigation. In *arxiv preprint arXiv:2306.14846*, 2023. 3, 18, 19, 20, 28

[30] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. A generalist agent. In *Transactions on Machine Learning Research (TMLR)*, November 10, 2022. 3, 12, 16, 18, 19, 22, 25

[31] Danny Driess, F. Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Ho Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Peter R. Florence. PaLM-E: An embodied multimodal language model. *ArXiv*, abs/2303.03378, 2023. 3, 12, 17, 18, 19, 24, 26

[32] Jean Kaddour, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, and Robert McHardy. Challenges and applications of large language models. *arXiv:2307.10169*, 2023. 3, 4

[33] Chenshuang Zhang, Chaoning Zhang, Mengchun Zhang, and In So Kweon. Text-to-image diffusion models in generative ai: A survey. *arXiv:2303.07909*, 2023.

[34] Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Bing Yin, and Xia Hu. Harnessing the power of llms in practice: A survey on chatgpt and beyond. *arXiv:2304.13712*, 2023. 4

[35] Sherry Yang, Ofir Nachum, Yilun Du, Jason Wei, Pieter Abbeel, and Dale Schuurmans. Foundation models for decision making: Problems, methods, and opportunities. *arXiv:2303.04129*, 2023. 3, 4

[36] Chaoning Zhang, Fachrina Dewi Puspitasari, Sheng Zheng, Chenghao Li, Yu Qiao, Taegoo Kang, Xinru Shan, Chenshuang Zhang, Caiyan Qin, Francois Rameau, Lik-Hang Lee, Sung-Ho Bae, and Choong Seon Hong. A survey on segment anything model (sam): Vision foundation model meets prompt engineering, 2023. 4

[37] Muhammad Awais, Muzammal Naseer, Salman Khan, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Foundational models defining a new era in vision: A survey and outlook, 2023. 4

[38] Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. A survey of vision-language pre-trained models. *IJCAI-2022 survey track*, 2022. 4

[39] Jindong Gu, Zhen Han, Shuo Chen, Ahmad Beirami, Bailan He, Ruotong Liao Gengyuan Zhang, Yao Qin, Volker Tresp, and Philip Torr. A systematic survey of prompt engineering on vision-language foundation models. *arXiv:2307.12980*, 2023. 4

[40] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Ji-Rong Wen. A survey on large language model based autonomous agents. *arXiv:2308.11432*, 2023. 4

[41] Jinzhou Lin, Han Gao, Rongtao Xu, Changwei Wang, Man Zhang, Li Guo, and Shibiao Xu. The development of llms for embodied navigation. In *IEEE/ASME TRANSACTIONS ON MECHATRONICS*, volume 1, Sept. 2023. 4

[42] Anirudha Majumdar. Robotics: An idiosyncratic snapshot in the age of llms, 8 2023. 4

[43] Xuan Xiao, Jiahang Liu, Zhipeng Wang, Yanmin Zhou, Yong Qi, Qian Cheng, Bin He, and Shuo Jiang. Robot learning in the era of foundation models: A survey, 2023. 4

[44] Roya Firoozi, Johnathan Tucker, Stephen Tian, Anirudha Majumdar, Jiankai Sun, Weiyu Liu, Yuke Zhu, Shuran Song, Ashish Kapoor, Karol Hausman, Brian Ichter, Danny Driess, Jiajun Wu, Cewu Lu, and Mac Schwager. Foundation models in robotics: Applications, challenges, and the future, 2023. 4

[45] Vincent Vanhoucke. The end-to-end false dichotomy: Roboticists arguing lego vs. playmo. *Medium*, October 28 2018. 5, 27

[46] Yuke Zhu. Cs391r: Robot learning, 2021. 5

[47] Jonathan Francis, Nariaki Kitamura, Felix Labelle, Xiaopeng Lu, Ingrid Navarro, and Jean Oh. Core challenges in embodied vision-language planning. *Journal of Artificial Intelligence Research*, 74:459–515, 2022. 5, 9

[48] Shaoqing Ren, Kaiming He, Ross Girshick, , and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 5

[49] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[50] Nikhil Varma Keetha, Chen Wang, Yuheng Qiu, Kuan Xu, and Sebastian Scherer. Airobject: A temporally evolving graph embedding for object identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8407–8416, 2022.

[51] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS*, 2017.

[52] Antoni Rosinol, John J Leonard, and Luca Carlone. Nerf-slam: Real-time dense monocular slam with neural radiance fields. *arXiv preprint arXiv:2210.13641*, 2022. 5

[53] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 5, 7, 8, 13

[54] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015. 5

[55] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *IEEE transactions on pattern analysis and machine intelligence*, 40(3):611–625, 2017.

[56] Xiang Gao, Rui Wang, Nikolaus Demmel, and Daniel Cremers. Ldso: Direct sparse odometry with loop closure. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2198–2204. IEEE, 2018.

[57] Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *European conference on computer vision*, pages 834–849. Springer, 2014.

[58] Jon Zubizarreta, Iker Aguinaga, and J. M. M. Montiel. Direct sparse mapping. *IEEE Transactions on Robotics*, 2020.

[59] Shibo Zhao, Peng Wang, Hengrui Zhang, Zheng Fang, and Sebastian Scherer. Tp-tio: A robust thermal-inertial odometry with deep thermalpoint. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4505–4512. IEEE, 2020. 5

[60] Wolfgang Hess, Damon Kohler, Holger Rapp, and Daniel Andor. Real-time loop closure in 2d lidar slam. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 1271–1278. IEEE, 2016. 5

[61] S. Kohlbrecher, J. Meyer, O. von Stryk, and U. Klingauf. A flexible and scalable slam system with full 3d motion estimation. In *Proc. IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*. IEEE, November 2011.

[62] Ji Zhang and Sanjiv Singh. Loam: Lidar odometry and mapping in real-time. In *Robotics: Science and systems*, volume 2, pages 1–9. Berkeley, CA, 2014. 5

[63] Shuo Yang, Zixin Zhang, Zhengyu Fu, and Zachary Manchester. Cerberus: Low-drift visual-inertial-leg odometry for agile locomotion. *ICRA*, 2023. 5

[64] Christian Forster, Luca Carlone, Frank Dellaert, and Davide Scaramuzza. Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation. Technical report, EPFL, 2015. 5

[65] Ji Zhang and Sanjiv Singh. Visual-lidar odometry and mapping: Low-drift, robust, and fast. In *ICRA*, 2015. 5

[66] Johannes Graeter, Alexander Wilczynski, and Martin Lauer. Limo: Lidar-monocular visual odometry. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 7872–7879. IEEE, 2018.

[67] Thien-Minh Nguyen, Shenghai Yuan, Muqing Cao, Thien Hoang Nguyen, and Lihua Xie. Viral slam: Tightly coupled camera-imu-uwb-lidar slam. *arXiv preprint arXiv:2105.03296*, 2021.

[68] Tixiao Shan, Brendan Englot, Drew Meyers, Wei Wang, Carlo Ratti, and Daniela Rus. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. In *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 5135–5142. IEEE, 2020.

[69] Shibo Zhao, Hengrui Zhang, Peng Wang, Lucas Nogueira, and Sebastian Scherer. Super odometry: Imu-centric lidar-visual-inertial estimator for challenging environments. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8729–8736. IEEE, 2021. 5

[70] Sen Wang, Ronald Clark, Hongkai Wen, and Niki Trigoni. Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 2043–2050. IEEE, 2017. 5

[71] Wenshan Wang, Yaoyu Hu, and Sebastian Scherer. Tartanvo: A generalizable learning-based vo. In *CoRL*, 2020. 5

[72] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017. 5

[73] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. In *NeurIPS*, 2021.

[74] Zihan Zhu, Songyou Peng, Viktor Larsson, Zhaopeng Cui, Martin R Oswald, Andreas Geiger, and Marc Pollefeys. Nicer-slam: Neural implicit scene encoding for rgb slam. *arXiv preprint arXiv:2302.03594*, 2023.

[75] Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. Splatam: Splat, track & map 3d gaussians for dense rgb-d slam. *arXiv preprint arXiv:2312.02126*, 2023. 5

[76] Lerrel Pinto, Dhiraj Gandhi, Yuanfeng Han, Yong-Lae Park, and Abhinav Gupta. The curious robot: Learning visual representations via physical interactions. In *ECCV*, 2016. 5

[77] Dinesh Jayaraman and Kristen Grauman. Learning to look around: Intelligently exploring unseen environments for unknown tasks. In *CVPR*, 2018. 5

[78] Liren Jin, Xieyuanli Chen, Julius Rückin, and Marija Popović. Neu-nbv: Next best view planning using uncertainty estimation in image-based neural rendering. *arXiv preprint arXiv:2303.01284*, 2023.

[79] Yafei Hu, Junyi Geng, Chen Wang, John Keller, and Sebastian Scherer. Off-Policy Evaluation with Online Adaptation for Robot Exploration in Challenging Environments. In *IEEE Robotics and Automation Letters (RA-L)*, 2023. 5

[80] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. 5

[81] Venkatraman Narayanan, Mike Phillips, and Maxim Likhachev. Anytime safe interval path planning for dynamic environments. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4708–4715, 2012.

[82] Sandip Aine, Siddharth Swaminathan, Venkatraman Narayanan, Victor Hwang, and Maxim Likhachev. Multi-heuristic A. In Dieter Fox, Lydia E. Kavraki, and Hanna Kurniawati, editors, *Robotics: Science and Systems X, University of California, Berkeley, USA, July 12-16, 2014*, 2014.

[83] Brian MacAllister, Jonathan Butzke, Alex Kushleyev, Harsh Pandey, and Maxim Likhachev. Path planning for non-circular micro aerial vehicles in constrained environments. In *2013 IEEE International Conference on Robotics and Automation*, pages 3933–3940, 2013.

[84] Benjamin J. Cohen, Sachin Chitta, and Maxim Likhachev. Single- and dual-arm motion planning with heuristic search. *Int. J. Robotics Res.*, 33(2):305–320, 2014. 5

[85] Steven M LaValle et al. Rapidly-exploring random trees: A new tool for path planning. Technical report, Iowa State University, 1998. 6

[86] J.J. Kuffner and S.M. LaValle. Rrt-connect: An efficient approach to single-query path planning. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, volume 2, pages 995–1001 vol.2, 2000.

[87] L.E. Kavraki, P. Svestka, J.-C. Latombe, and M.H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, 1996.

[88] Sertac Karaman and Emilio Frazzoli. Sampling-based algorithms for optimal motion planning. *The international journal of robotics research*, 30(7):846–894, 2011.

[89] Jonathan D Gammell, Siddhartha S Srinivasa, and Timothy D Barfoot. Batch informed trees (bit): Sampling-based optimal planning via the heuristically guided search of implicit random geometric graphs. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 3067–3074. IEEE, 2015.

[90] Sanjiban Choudhury, Jonathan D Gammell, Timothy D Barfoot, Siddhartha S Srinivasa, and Sebastian Scherer. Regionally accelerated batch informed trees (rabit): A framework to integrate local information into optimal path planning. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4207–4214. IEEE, 2016. 6

[91] Malik Ghallab, Dana Nau, and Paolo Traverso. *Automated Planning and Acting*. Cambridge University Press, 2016. 6

[92] Caelan Reed Garrett, Rohan Chitnis, Rachel Holladay, Beomjoon Kim, Tom Silver, Leslie Pack Kaelbling, and Tomas Lozano-P´erez. Integrated Task and Motion Planning. In *arXiv:2010.01083*, 2010. 6, 10

[93] Dhruv Shah, Arjun Bhorkar, Hrish Leen, Ilya Kostrikov, Nick Rhinehart, and Sergey Levine. Offline reinforcement learning for visual navigation. In *CoRL*, 2022. 6

[94] Kyle Stachowicz, Arjun Bhorkar, Dhruv Shah, Ilya Kostrikov, and Sergey Levine. Fastrlap: A system for learning high-speed driving via deep rl and autonomous practicing. *arXiv pre-print*, 2023. 6

[95] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. In *NeurIPS 2020 Tutorial*, 2020. 6

[96] Fangkai Yang, Daoming Lyu, Bo Liu, and Steven Gustafson. Peorl: Integrating symbolic planning and hierarchical reinforcement learning for robust decision-making. *arXiv preprint arXiv:1804.07779*, 2018. 6

[97] Yuqian Jiang, Fangkai Yang, Shiqi Zhang, and Peter Stone. Task-motion planning with reinforcement learning for adaptable mobile service robots. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7529–7534. IEEE, 2019. 6

[98] Garrett Andersen and George Konidaris. Active exploration for learning symbolic representations. *Advances in Neural Information Processing Systems*, 30, 2017. 6

[99] George Konidaris, Leslie Pack Kaelbling, and Tomas Lozano-Perez. From skills to symbols: Learning symbolic representations for abstract high-level planning. *Journal of Artificial Intelligence Research*, 61:215–289, 2018. 6

[100] Brian Paden, Michal Čáp, Sze Zheng Yong, Dmitry Yershov, and Emilio Frazzoli. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on intelligent vehicles*, 1(1):33–55, 2016. 6

[101] Jonathan Francis, Bingqing Chen, Weiran Yao, Eric Nyberg, and Jean Oh. Distribution-aware goal prediction and conformant model-based planning for safe autonomous driving. *ICML Workshop on Safe Learning for Autonomous Driving*, 2022. 6

[102] James Herman, Jonathan Francis, Siddha Ganju, Bingqing Chen, Anirudh Koul, Abhinav Gupta, Alexey Skabelkin, Ivan Zhukov, Max Kumskoy, and Eric Nyberg. Learn-to-race: A multimodal control environment for autonomous racing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9793–9802, 2021. 9, 22

[103] Jonathan Francis, Bingqing Chen, Siddha Ganju, Sidharth Kathpal, Jyotish Poonganam, Ayush Shivani, Vrushank Vyas, Sahika Genc, Ivan Zhukov, Max Kumskoy, et al. Learn-to-race challenge 2022: Benchmarking safe learning and cross-domain generalisation in autonomous racing. *ICML Workshop on Safe Learning for Autonomous Driving*, 2022. 6, 9

[104] Grady Williams, Andrew Aldrich, and Evangelos A Theodorou. Model predictive path integral control: From theory to parallel computation. *Journal of Guidance, Control, and Dynamics*, 40(2):344–357, 2017. 6

[105] Christopher G Atkeson and Stefan Schaal. Robot learning from demonstration. In *ICML*, 1997. 6

[106] Jens Kober, J. Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 2013. 6

[107] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 6

[108] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. In *NIPS Deep Learning Workshop*, 2013. 6

[109] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. In *Nature*, 2016. 6

[110] Elia Kaufmann, Antonio Loquercio, René Ranftl, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Deep drone acrobatics. In *Proceedings of Robotics: Science and Systems*, Corvalis, Oregon, USA, July 2020. 6

[111] Sergey Levine, Peter Pastor, Alex Krizhevsky, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. In *arXiv:1603.02199*, 2016. 6, 9

[112] Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, and Sergey Levine. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. In *CoRL*, 2018. 9, 10, 18

[113] Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. In *Science Robotics*, 21 Oct 2020. 6

[114] Stephane Ross, Geoffrey J. Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *AISTATS*, 2011. 6

[115] Pieter Abbeel and Andrew Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *ICML*, 2004. 6

[116] Brian D. Ziebart, Andrew Maas, J.Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, 2008. 6

[117] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. In *NIPS*, 2014. 6

[118] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *NIPS*, 2016. 6

[119] Yunpeng Pan, Ching-An Cheng, Kamil Saigol, Keuntaek Lee, Xinyan Yan, Evangelos Theodorou, and Byron Boots. Agile autonomous driving using end-to-end deep imitation learning. In *RSS*, 2018. 6

[120] Chenhao Li, Marin Vlastelica, Sebastian Blaes, Jonas Frey, Felix Grimminger, and Georg Martius. Learning agile skills via adversarial imitation of rough partial demonstrations. In *CoRL*, 2022. 6

[121] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction , second edition*. The MIT Press, 2018. 6

[122] Bingqing Chen, Jonathan Francis, Jean Oh, Eric Nyberg, and Sylvia L Herbert. Safe autonomous racing via approximate reachability on ego-vision. *arXiv preprint arXiv:2110.07699*, 2021. 6, 11, 26, 27

[123] Deepak Pathak Zipeng Fu, Xuxin Cheng. Deep whole-body control: Learning a unified policy for manipulation and locomotion. In *CoRL*, 2022. 6

[124] Xuxin Cheng, Kexin Shi, Ananye Agarwal, and Deepak Pathak. Extreme parkour with legged robots. In *arXiv:2309.14341*, 2023. 6

[125] Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Neural Information Processing Systems*, 2018. 6

[126] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *NIPS*, 2016. 6

[127] Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *ICRA*, 2017. 6

[128] S. Levine I. Kostrikov, A. Nair. Offline reinforcement learning with implicit q-learning. In *ICLR*, 2022. 6

[129] Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y. Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. In *NeurIPS*, 2020. 6

[130] Wenxuan Zhou, Sujay Bajracharya, and David Held. Plas: Latent action space for offline reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2020. 6

[131] Rafael Rafailov, Tianhe Yu, Aravind Rajeswaran, and Chelsea Finn. Offline reinforcement learning from images with latent space models. In *Proceedings of Machine Learning Research*, volume 144:1–15, 2021. 6

[132] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollar, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 7

[133] Namuk Park, Wonjae Kim, Byeongho Heo, Taekyung Kim, and Sangdoo Yun. What do self-supervised vision transformers learn? In *ICLR*, 2023. 7

[134] Shashank Shekhar, Florian Bordes, Pascal Vincent, and Ari Morcos. Objectives matter: Understanding the impact of self-supervised objectives on vision transformer representations. *arXiv:2304.13089*, 2023. 7

[135] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, et al. Conceptfusion: Open-set multimodal 3d mapping. *RSS*, 2023.

[136] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv:2112.05814*, 2021.

[137] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2955–2966, 2023. 7

[138] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62, 2022. 7, 28

[139] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. *arXiv preprint arXiv:2211.05778*, 2022. 7, 8

[140] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020. 7

[141] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15619–15629, 2023. 7

[142] Adrien Bardes, Jean Ponce, and Yann LeCun. Mc-jepa: A joint-embedding predictive architecture for self-supervised learning of motion and content features. *arXiv preprint arXiv:2307.12698*, 2023. 7

[143] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2006. 7

[144] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 7

[145] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. Technical report, OpenAI, 2018. 7

[146] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019. 8

[147] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. 8

[148] Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Bing Yin, and Xia Hu. Harnessing the power of llms in practice: A survey on chatgpt and beyond, 2023. 8

[149] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017. 8

[150] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *ICLR*, 2022. 8

[151] Boyi Li, Kilian Q. Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 8, 13

[152] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *arXiv:1908.02265*, 2019. 8

[153] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, 2021. 8

[154] Hangbo Bao, Wenhui Wang, Li Dong, and Furu Wei. Vl-beit: Generative vision-language pretraining, 2022. 8

[155] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning. *ArXiv*, abs/2204.14198, 2022. 8

[156] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. GIT: A generative image-to-text transformer for vision and language. *Transactions on Machine Learning Research*, 2022. 8

[157] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, 2022. 8

[158] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. *CoRR*, abs/2301.12597, 2023. 8

[159] OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023. 8

[160] Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm, 2023. 8

[161] Rongjie Huang, Mingze Li, Dongchao Yang, Jiatong Shi, Xuankai Chang, Zhenhui Ye, Yuning Wu, Zhiqing Hong, Jiawei Huang, Jinglin Liu, Yi Ren, Zhou Zhao, and Shinji Watanabe. Audiogpt: Understanding and generating speech, music, sound, and talking head, 2023. 8

[162] Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities, 2023. 8

[163] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding, 2023. 8

[164] Peide Huang, Xilun Zhang, Ziang Cao, Shiqi Liu, Mengdi Xu, Wenhao Ding, Jonathan Francis, Bingqing Chen, and Ding Zhao. What went wrong? closing the sim-to-real gap via differentiable causal discovery. In *7th Annual Conference on Robot Learning*, 2023. 9, 22

[165] Jonathan Francis. *Knowledge-enhanced Representation Learning for Multiview Context Understanding*. PhD thesis, Carnegie Mellon University, 2022. 9

[166] Gyan Tatiya, Jonathan Francis, and Jivko Sinapov. Transferring implicit knowledge of non-visual object properties across heterogeneous robot morphologies. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11315–11321. IEEE, 2023. 9, 14

[167] Gyan Tatiya, Jonathan Francis, and Jivko Sinapov. Cross-tool and cross-behavior perceptual knowledge transfer for grounded object recognition. *arXiv preprint arXiv:2303.04023*, 2023. 9, 14

[168] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous

driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 9

[169] Alexander Herzog*, Kanishka Rao*, Karol Hausman*, Yao Lu*, Paul Wohlhart*, Mengyuan Yan, Jessica Lin, Montserrat Gonzalez Arenas, Ted Xiao, Daniel Kappler, Daniel Ho, Jarek Rettinghouse, Yevgen Chebotar, Kuang-Huei Lee, Keerthana Gopalakrishnan, Ryan Julian, Adrian Li, Chuyuan Kelly Fu, Bob Wei, Sangeetha Ramesh, Khem Holden, Kim Kleiven, David Rendleman, Sean Kirmani, Jeff Bingham, Jon Weisz, Ying Xu, Wenlong Lu, Matthew Bennice, Cody Fong, David Do, Jessica Lam, Yunfei Bai, Benjie Holson, Michael Quinlan, Noah Brown, Mrinal Kalakrishnan, Julian Ibarz, Peter Pastor, and Sergey Levine. Deep rl at scale: Sorting waste in office buildings with a fleet of mobile manipulators. In *Robotics: Science and Systems (RSS)*, 2023. 9

[170] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012. 9, 22, 25

[171] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. Isaac gym: High performance gpu-based physics simulation for robot learning, 2021. 22, 25

[172] Mayank Mittal, Calvin Yu, Qinxi Yu, Jingzhou Liu, Nikita Rudin, David Hoeller, Jia Lin Yuan, Pooria Poorsarvi Tehrani, Ritvik Singh, Yunrong Guo, Hammad Mazhar, Ajay Mandlekar, Buck Babich, Gavriel State, Marco Hutter, and Animesh Garg. Orbit: A unified simulation framework for interactive robot learning environments, 2023.

[173] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019.

[174] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: Training home assistants to rearrange their habitat, 2022. 21

[175] Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander William Clegg, Michal Hlavac, So Yeon Min, Vladimír Vondruš, Theophile Gervet, Vincent-Pierre Berges, John M. Turner, Oleksandr Maksymets, Zsolt Kira, Mrinal Kalakrishnan, Jitendra Malik, Devendra Singh Chaplot, Unnat Jain, Dhruv Batra, Akshara Rai, and Roozbeh Mottaghi. Habitat 3.0: A co-habitat for humans, avatars and robots, 2023. 9, 21

[176] Embodiment Collaboration. Open x-embodiment: Robotic learning datasets and rt-x models, 2023. 9, 18, 20, 21, 22

[177] Elena Arcari, Maria Vittoria Minniti, Anna Scampicchio, Andrea Carron, Farbod Farshidian, Marco Hutter, and Melanie N. Zeilinger. Bayesian multi-task learning mpc for robotic mobile manipulation, 2023. 10

[178] Chao Cao, Hongbiao Zhu, Howie Choset, and Ji Zhang. TARE: A Hierarchical Framework for Efficiently Exploring Complex 3D Environments. In *ICRA*, 2023. 10

[179] Fahad Islam, Oren Salzman, Aditya Agarwal, and Maxim Likhachev. Provably constant-time planning and replanning for real-time grasping objects off a conveyor belt. In *RSS*, 2020. 10

[180] Dhruv Mauria Saxena, Muhammad Suhail Saleem, and Maxim Likhachev. Manipulation planning among movable obstacles using physics-based adaptive motion primitives. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2021. 10

[181] Yuchen Cui, Scott Niekum, Abhinav Gupta, Vikash Kumar, and Aravind Rajeswaran. Can foundation models perform zero-shot task specification for robot manipulation? In *Learning for Dynamics and Control Conference*, pages 893–905. PMLR, 2022. 11, 20

[182] Yecheng Jason Ma, William Liang, Guanzhi Wang, De-An Huang, Osbert Bastani, Dinesh Jayaraman, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Eureka: Human-level reward design via coding large language models. In *2nd Workshop on Language and Robot Learning: Language as Grounding*, 2023. 11, 15, 20, 25, 26

[183] Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, et al. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, 56(Suppl 1):1513–1589, 2023. 11

[184] Xuhong Li, Haoyi Xiong, Xingjian Li, Xuanyu Wu, Xiao Zhang, Ji Liu, Jiang Bian, and Dejing Dou. Interpretable deep learning: Interpretation, interpretability, trustworthiness, and beyond. *Knowledge and Information Systems*, 64(12):3197–3234, 2022. 11

[185] Ta-Chung Chi, Minmin Shen, Mihail Eric, Seokhwan Kim, and Dilek Hakkani-tur. Just ask: An interactive learning framework for vision and language navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2459–2466, 2020. 11

[186] Anastasios N Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*, 2021. 11

[187] Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners. In *7th Annual Conference on Robot Learning*, 2023. 11, 20, 27

[188] Kim P Wabersich, Andrew J Taylor, Jason J Choi, Koushil Sreenath, Claire J Tomlin, Aaron D Ames, and Melanie N Zeilinger. Data-driven safety filters: Hamilton-jacobi reachability, control barrier functions, and predictive methods for uncertain systems. *IEEE Control Systems Magazine*, 43(5):137–177, 2023. 11

[189] Kai-Chieh Hsu, Haimin Hu, and Jaime Fernández Fisac. The safety filter: A unified view of safety-critical control in autonomous systems. *arXiv preprint arXiv:2309.05837*, 2023. 11

[190] Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European control conference (ECC)*, pages 3420–3431. IEEE, 2019. 11

[191] Somil Bansal, Mo Chen, Sylvia Herbert, and Claire J Tomlin. Hamilton-jacobi reachability: A brief overview and recent advances. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 2242–2253. IEEE, 2017. 11

[192] Karen Leung, Nikos Aréchiga, and Marco Pavone. Backpropagation through signal temporal logic specifications: Infusing logical structure into gradient-based methods. *The International Journal of Robotics Research*, 42(6):356–370, 2023. 11

[193] Shangding Gu, Long Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, Yaodong Yang, and Alois Knoll. A review of safe reinforcement learning: Methods, theory and applications. *arXiv preprint arXiv:2205.10330*, 2022. 11

[194] Charles Dawson, Sicun Gao, and Chuchu Fan. Safe control with learned certificates: A survey of neural lyapunov, barrier, and contraction methods for robotics and control. *IEEE Transactions on Robotics*, 2023. 11

[195] Konstantinos Bousmalis, Giulia Vezzani, Dushyant Rao, Coline Devin, Alex X. Lee, Maria Bauza, Todor Davchev, Yuxiang Zhou, Agrim Gupta, Akhil Raju, Antoine Laurens, Claudio Fantacci, Valentin Dalibard, Martina Zambelli, Murilo Martins, Rugile Pevceviciute, Michiel Blokzijl, Misha Denil, Nathan Batchelor, Thomas Lampe, Emilio Parisotto, Konrad Żo lna, Scott Reed, Sergio Gómez Colmenarejo, Jon Scholz, Abbas Abdolmaleki, Oliver Groth, Jean-Baptiste Regli, Oleg Sushkov, Tom Rothörl, José Enrique Chen, Yusuf Aytar, Dave Barker, Joy Ortiz, Martin Riedmiller, Jost Tobias Springenberg, Raia Hadsell, Francesco Nori, and Nicolas Heess. Robocat: A self-improving foundation agent for robotic manipulation, 2023. 12, 18, 20, 24

[196] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *CoRL*, 2021. 12, 13

[197] William Shen, Ge Yang, Alan Yu, Jansen Wong, Leslie Pack Kaelbling, and Phillip Isola. Distilled feature fields enable few-shot language-guided manipulation. *CoRL*, 2023. 13, 23

[198] Yanjie Ze, Ge Yan, Yueh-Hua Wu, Annabella Macaluso, Yuying Ge, Jianglong Ye, Nicklas Hansen, Li Erran Li, and Xiaolong Wang. Multi-task real robot learning with generalizable neural feature fields. *CoRL*, 2023. 12, 13, 23

[199] Reihaneh Mirjalili, Michael Krawez, and Wolfram Burgard. Fm-loc: Using foundation models for improved vision-based localization. *arXiv:2304.07058*, 2023. 13

[200] Gyan Tatiya, Jonathan Francis, Ho-Hsiang Wu, Yonatan Bisk, and Jivko Sinapov. Mosaic: Learning unified multi-sensory object property representations for robot perception. *arXiv preprint arXiv:2309.08508*, 2023. 13, 14, 21

[201] Chenguang Huang, Oier Mees, Andy Zeng, and Wolfram Burgard. Visual language maps for robot navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, London, UK, 2023. 12, 13, 16, 22

[202] Nur Muhammad (Mahi) Shafiullah, Chris Paxton, Lerrel Pinto1 Soumith Chintala, and Arthur Szlam. Clip-fields: Weakly supervised semantic fields for robotic memory. In *RSS*, 2023. 13, 16, 25

[203] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Shuang Li, Ganesh Iyer, Soroush Saryazd, Nikhil Keetha, Ayush Tewari, Joshua B. Tenenbaum, Celso Miguel de Melo, Madhava Krishna, Liam Paull, Florian Shkurti, and Antonio Torralba. Conceptfusion: Open-set multimodal 3d mapping. In *arXiv:2302.07241*, 2023. 13

[204] Dhruv Shah, Blazej Osinski, Brian Ichter, and Sergey Levine. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *CoRL*, 2022. 12, 13, 25

[205] Sriram Yenamandra, Arun Ramachandran, Mukul Khanna, Karmesh Yadav, Devendra Singh Chaplot, Gunjan Chhablani, Alexander Clegg, Theophile Gervet, Vidhi Jain, Ruslan Partsey, Ram Ramrakhya, Andrew Szot, Tsung-Yen Yang, Aaron Edsinger, Charlie Kemp, Binit Shah, Zsolt Kira, Dhruv Batra, Roozbeh Mottaghi, Yonatan Bisk, and Chris Paxton. The homerobot open vocab mobile manipulation challenge. In *Thirty-seventh Conference on Neural Information Processing Systems: Competition Track*, 2023. 12, 24, 29

[206] Theophile Gervet, Zhou Xian, Nikolaos Gkanatsios, and Katerina Fragkiadaki. Act3d: 3d feature field transformers for multi-task robotic manipulation, 2023. 13

[207] Christina Kassab, Matias Mattamala, Lintong Zhang, and Maurice Fallon. Language-extended indoor slam (lexis): A versatile system for real-time visual scene understanding. *arXiv preprint arXiv:2309.15065*, 2023. 13

[208] Nikhil Keetha, Avneesh Mishra, Jay Karhade, Krishna Murthy Jatavallabhula, Sebastian Scherer, Madhava Krishna, and Sourav Garg. Anyloc: Towards universal visual place recognition. *RA-L*, 2023. 13

[209] Yao He, Ivan Cisneros, Nikhil Keetha, Jay Patrikar, Zelin Ye, Ian Higgins, Yaoyu Hu, Parv Kapoor, and Sebastian Scherer. Foundloc: Vision-based onboard aerial localization in the wild, 2023. 13

[210] Jivko Sinapov, Connor Schenck, Kerrick Staley, Vladimir Sukhoy, and Alexander Stoytchev. Grounding semantic categories in behavioral interactions: Experiments with 100 objects. *Robotics and Autonomous Systems*, 62(5):632–645, may 2014. 14

[211] Jivko Sinapov, Connor Schenck, and Alexander Stoytchev. Learning relational object categories using behavioral exploration and multimodal perception. In *International Conference on Robotics and Automation (ICRA)*, pages 5691–5698, Hong Kong, China, may 2014. IEEE.

[212] Mevlana C. Gemici and Ashutosh Saxena. Learning haptic representation for manipulating deformable food objects. In *Intelligent Robots and Systems (IROS)*, pages 638–645, Chicago, IL, USA, Sep 2014. IEEE.

[213] Gyan Tatiya, Ramtin Hosseini, Michael C. Hughes, and Jivko Sinapov. Sensorimotor cross-behavior knowledge transfer for grounded category recognition. In *International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. IEEE, 2019.

[214] Gyan Tatiya, Ramtin Hosseini, Michael Hughes, and Jivko Sinapov. A framework for sensorimotor cross-perception and cross-behavior knowledge transfer for object categorization. *Frontiers in Robotics and AI*, 7:137, 2020.

[215] Gyan Tatiya, Yash Shukla, Michael Edegware, and Jivko Sinapov. Haptic knowledge transfer between heterogeneous robots using kernel manifold alignment. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020. 14

[216] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. *ArXiv*, abs/2209.07753, 2023. 14, 24, 26

[217] Yilun Du, Mengjiao Yang, Pete Florence, Fei Xia, Ayzaan Wahid, Brian Ichter, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B Tenenbaum, et al. Video language planning. *arXiv preprint arXiv:2310.10625*, 2023. 14

[218] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683, 2018. 14

[219] Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents, 2022. 14, 16

[220] Kevin Lin, Christopher Agia, Toki Migimatsu, Marco Pavone, and Jeannette Bohg. Text2motion: From natural language instructions to feasible plans. *arXiv preprint arXiv:2303.12153*, 2023. 14, 23

[221] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. ProgPrompt: Generating situated robot task plans using large language models, 2022. 14, 16, 26

[222] Lirui Wang, Yiyang Ling, Zhecheng Yuan, Mohit Shridhar, Chen Bao, Yuzhe Qin, Bailin Wang, Huazhe Xu, and Xiaolong Wang. Gensim: Generating robotic simulation tasks via large language models. In *CoRL*, 2023. 14, 17, 23

[223] J. Seipp, Á. Torralba, and J. Hoffmann. Pddl generators, 2022. 15, 24

[224] Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. Sayplan: Grounding large language models using 3d scene graphs for scalable task planning. In *7th Annual Conference on Robot Learning*, 2023. 15

[225] Quanting Xie, Tianyi Zhang, Kedi Xu, Matthew Johnson-Roberson, and Yonatan Bisk. Reasoning about the unseen for efficient outdoor object navigation, 2023. 15, 17, 22, 28

[226] Wenhao Yu, Nimrod Gileadi, Chuyuan Fu, Sean Kirmani, Kuang-Huei Lee, Montse Gonzalez Arenas, Hao-Tien Lewis Chiang, Tom Erez, Leonard Hasenclever, Jan Humplik, Brian Ichter, Ted Xiao, Peng Xu, Andy Zeng, Tingnan Zhang, Nicolas Heess, Dorsa Sadigh, Jie Tan, Yuval Tassa, and Fei Xia. Language to rewards for robotic skill synthesis. *Arxiv preprint arXiv:2306.08647*, 2023. 15, 20, 25, 26

[227] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023. 15, 16, 22, 23

[228] Junting Chen, Guohao Li, Suryansh Kumar, Bernard Ghanem, and Fisher Yu. How to not train your dragon: Training-free embodied object goal navigation with semantic frontiers. *arXiv preprint arXiv:2305.16925*, 2023. 15

[229] Yen-Jen Wang, Bike Zhang, Jianyu Chen, and Koushil Sreenath. Prompt a robot to walk with large language models, 2023. 15, 25

[230] Yuqing Du, Olivia Watkins, Zihan Wang, Cédric Colas, Trevor Darrell, Pieter Abbeel, Abhishek Gupta, and Jacob Andreas. Guiding pretraining in reinforcement learning with large language models, 2023. 15

[231] Minae Kwon, Sang Michael Xie, Kalesha Bullard, and Dorsa Sadigh. Reward design with language models, 2023.

[232] Tianbao Xie, Siheng Zhao, Chen Henry Wu, Yitao Liu, Qian Luo, Victor Zhong, Yanchao Yang, and Tao Yu. Text2reward: Automated dense reward function generation for reinforcement learning, 2023. 15, 26

[233] Wenlong Huang, Fei Xia, Dhruv Shah, Danny Driess, Andy Zeng, Yao Lu, Pete Florence, Igor Mordatch, Sergey Levine, Karol Hausman, and Brian Ichter. Grounded decoding: Guiding text generation with grounded models for robot control, 2023. 16

[234] Thomas Carta, Clément Romac, Thomas Wolf, Sylvain Lamprier, Olivier Sigaud, and Pierre-Yves Oudeyer. Grounding large language models in interactive environments with online reinforcement learning, 2023. 16, 17

[235] Yen-Jen Wang, Bike Zhang, Jianyu Chen, and Koushil Sreenath. Prompt a robot to walk with large language models, 2023. 16

[236] Huy Ha, Pete Florence, and Shuran Song. Scaling up and distilling down: Language-guided robot skill acquisition. In *Proceedings of the 2023 Conference on Robot Learning*, 2023. 17, 20, 23

[237] Yufei Wang, Zhou Xian, Feng Chen, Tsun-Hsuan Wang, Yian Wang, Katerina Fragkiadaki, Zackory Erickson, David Held, and Chuang Gan. Robogen: Towards unleashing infinite data for automated robot learning via generative simulation, 2023. 17, 20

[238] Tianhe Yu, Ted Xiao, Austin Stone, Jonathan Tompson, Anthony Brohan, Su Wang, Jaspiar Singh, Clayton Tan, Jodilyn Peralta Dee M, Brian Ichter, Karol Hausman, and Fei Xia. Scaling robot learning with semantically imagined experience. In *arXiv:2302.11550*, 2023. 17, 20

[239] Jiayuan Gu, Sean Kirmani, Paul Wohlhart, Yao Lu, Montserrat Gonzalez Arenas, Kanishka Rao, Wenhao Yu, Chuyuan Fu, Keerthana Gopalakrishnan, Zhuo Xu, Priya Sundaresan, Peng Xu, Hao Su, Karol Hausman, Chelsea Finn, Quan Vuong, and Ted Xiao. Rt-trajectory: Robotic task generalization via hindsight trajectory sketches, 2023. 17, 20

[240] Kevin Black, Mitsuhiko Nakamoto, Pranav Atreya, Homer Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. Zero-shot robotic manipulation with pretrained image-editing diffusion models, 2023. 17

[241] Yilun Du, Mengjiao Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Joshua B. Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *NeurIPS*, 2023. 17

[242] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. 17

[243] Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D. Hwang, Soumya Sanyal, Sean Welleck, Xiang Ren, Allyson Ettinger, Zaid Harchaoui, and Yejin Choi. Faith and fate: Limits of transformers on compositionality, 2023. 17

[244] Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Michal Podstawski, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. Graph of thoughts: Solving elaborate problems with large language models, 2023. 17

[245] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models, 2023. 17

[246] Shun Zhang, Zhenfang Chen, Yikang Shen, Mingyu Ding, Joshua B. Tenenbaum, and Chuang Gan. Planning with large language models for code generation. In *The Eleventh International Conference on Learning Representations*, 2023. 17

[247] Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. Llm+p: Empowering large language models with optimal planning proficiency, 2023. 17, 24

[248] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Noah Brown, Tomas Jackson, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models, 2022. 17, 24

[249] Austin Stone, Ted Xiao, Yao Lu, Keerthana Gopalakrishnan, Kuang-Huei Lee, Quan Vuong, Paul Wohlhart, Brianna Zitkovich, Fei Xia, Chelsea Finn, et al. Open-world object manipulation using pre-trained vision-language models. *arXiv preprint arXiv:2303.00905*, 2023. 18, 19, 23

[250] Shuang Li, Xavier Puig, Chris Paxton, Yilun Du, Clinton Wang, Linxi Fan, Tao Chen, De-An Huang, Ekin Akyürek, Anima Anandkumar, et al. Pre-trained language models for interactive decision-making. *Advances in Neural Information Processing Systems*, 35:31199–31212, 2022. 18

[251] Siddharth Karamcheti, Megha Srivastava, Percy Liang, and Dorsa Sadigh. Lila: Language-informed latent actions. In *Conference on Robot Learning*, pages 1379–1390. PMLR, 2022. 18, 20

[252] Hengyuan Hu and Dorsa Sadigh. Language instructed reinforcement learning for human-ai coordination. *arXiv preprint arXiv:2304.07297*, 2023. 18

[253] Corey Lynch, Ayzaan Wahid, Jonathan Tompson, Tianli Ding, James Betker, Robert Baruch, Travis Armstrong, and Pete Florence. Interactive language: Talking to robots in real time. *arXiv preprint arXiv:2210.06407*, 2022. 18, 21, 23, 24

[254] Kuang-Huei Lee, Ted Xiao, Adrian Li, Paul Wohlhart, Ian Fischer, and Yao Lu. Pi-qt-opt: Predictive information improves multi-task robotic reinforcement learning at scale. *CoRL*, 2022. 18

[255] Alexander Herzog, Kanishka Rao, Karol Hausman, Yao Lu, Paul Wohlhart, Mengyuan Yan, Jessica Lin, Montserrat Gonzalez Arenas, Ted Xiao, Daniel Kappler, Daniel Ho, Jarek Rettinghouse, Yevgen Chebotar, Kuang-Huei Lee, Keerthana Gopalakrishnan, Ryan Julian, Adrian Li, Chuyuan Kelly Fu, Bob Wei, Sangeetha Ramesh, Khem Holden, Kim Kleiven, David Rendleman, Sean Kirmani, Jeff Bingham, Jon Weisz, Ying Xu, Wenlong Lu, Matthew Bennice, Cody Fong, David Do, Jessica Lam, Yunfei Bai, Benjie Holson, Michael Quinlan, Noah Brown, Mrinal Kalakrishnan, Julian Ibarz, Peter Pastor, and Sergey Levine. Deep rl at scale: Sorting waste in office buildings with a fleet of mobile manipulators. In *RSS*, 2023. 18

[256] Yevgen Chebotar, Quan Vuong, Alex Irpan, Karol Hausman, Fei Xia, Yao Lu, Aviral Kumar, Tianhe Yu, Alexander Herzog, Karl Pertsch, Keerthana Gopalakrishnan, Julian Ibarz, Ofir Nachum, Sumedh Sontakke, Grecia Salazar, Huong T Tran, Jodilyn Peralta, Clayton Tan, Deeksha Manjunath, Jaspiar Singht, Brianna Zitkovich, Tomas Jackson, Kanishka Rao, Chelsea Finn, and Sergey Levine. Q-transformer: Scalable offline reinforcement learning via autoregressive q-functions. In *CoRL*, 2023. 19, 20

[257] Aviral Kumar, Anikait Singh, Frederik Ebert, Mitsuhiko Nakamoto, Yanlai Yang, Chelsea Finn, and Sergey Levine. Pre-training for robots: Offline rl enables learning new tasks from a handful of trials. In *RSS*, 2023. 19

[258] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. In *NeurIPS*, 2020. 19

[259] Simone Parisi, Aravind Rajeswaran, Senthil Purushwalkam, and Abhinav Gupta. The unsurprising effectiveness of pre-trained vision models for control, 2022. 19

[260] Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation, 2022. 24

[261] Ilija Radosavovic, Tete Xiao, Stephen James, Pieter Abbeel, Jitendra Malik, and Trevor Darrell. Real-world robot learning with masked visual pre-training. In *Conference on Robot Learning*, pages 416–426. PMLR, 2023. 19

[262] Ilija Radosavovic, Baifeng Shi, Letian Fu, Ken Goldberg, Trevor Darrell, and Jitendra Malik. Robot learning with sensorimotor pre-training, 2023. 19

[263] Nicklas Hansen, Zhecheng Yuan, Yanjie Ze, Tongzhou Mu, Aravind Rajeswaran, Hao Su, Huazhe Xu, and Xiaolong Wang. On pre-training for visuo-motor control: Revisiting a learning-from-scratch baseline. In *ICML*, 2023. 19

[264] Arjun Majumdar, Karmesh Yadav, Sergio Arnaud, Yecheng Jason Ma, Claire Chen, Sneha Silwal, Aryan Jain, Vincent-Pierre Berges, Pieter Abbeel, Jitendra Malik, Dhruv Batra, Yixin Lin, Oleksandr Maksymets, Aravind Rajeswaran, and Franziska Meier. Where are we in the search for an artificial visual cortex for embodied intelligence?, 2023. 19, 25

[265] Xi Chen, Josip Djolonga, Piotr Padlewski, Basil Mustafa, Soravit Changpinyo, Jialin Wu, Carlos Riquelme Ruiz, Sebastian Goodman, Xiao Wang, Yi Tay, Siamak Shakeri, Mostafa Dehghani, Daniel Salz, Mario Lucic, Michael Tschannen, Arsha Nagrani, Hexiang Hu, Mandar Joshi, Bo Pang, Ceslee Montgomery, Paulina Pietrzyk, Marvin Ritter, AJ Piergiovanni, Matthias Minderer, Filip Pavetic, Austin Waters, Gang Li, Ibrahim Alabdulmohsin, Lucas Beyer, Julien Amelot, Kenton Lee, Andreas Peter Steiner, Yang Li, Daniel Keysers, Anurag Arnab, Yuanzhong Xu, Keran Rong, Alexander Kolesnikov, Mojtaba Seyedhosseini, Anelia Angelova, Xiaohua Zhai, Neil Houlsby, and Radu Soricut. Pali-x: On scaling up a multilingual vision and language model, 2023. 19

[266] Shikhar Bahl, Russell Mendonca, Lili Chen, Unnat Jain, and Deepak Pathak. Affordances from human videos as a versatile representation for robotics. *CVPR*, 2023. 19

[267] Dhruv Shah, Ajay Sridhar, Arjun Bhorkar, Noriaki Hirose, and Sergey Levine. Gnm: A general navigation model to drive any robot. In *ICRA*, 2023. 19, 21, 25

[268] Joanne Truong, April Zitkovich, Sonia Chernova, Dhruv Batra, Tingnan Zhang, Jie Tan, and Wenhao Yu. Indoorsim-to-outdoorreal: Learning to navigate outdoors without any outdoor experience. In *arXiv:2305.01098*, 2023. 19

[269] Rogerio Bonatti, Sai Vemprala, Shuang Ma, Felipe Frujeri, Shuhang Chen, and Ashish Kapoor. Pact: Perception-action causal transformer for autoregressive robotics pre-training. In *arXiv:2209.11133*, 2022. 19

[270] Qiao Gu, Alihusein Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. *arXiv preprint arXiv:2309.16650*, 2023. 20

[271] Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. *arXiv preprint arXiv:2305.14992*, 2023. 20, 28

[272] Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. VIMA: general robot manipulation with multimodal prompts. *ArXiv*, abs/2210.03094, 2022. 20, 23

[273] Shikhar Bahl, Abhinav Gupta, and Deepak Pathak. Human-to-robot imitation in the wild. In *RSS*, 2022. 20

[274] Vidhi Jain, Yixin Lin, Eric Undersander, Yonatan Bisk, and Akshara Rai. Transformers are adaptable task planners. In *6th Annual Conference on Robot Learning*, 2022. 20

[275] Jacob Andreas, Dan Klein, and Sergey Levine. Modular multitask reinforcement learning with policy sketches. In *International conference on machine learning*, pages 166–175. PMLR, 2017. 20

[276] Marjorie Skubic, Derek Anderson, Samuel Blisard, Dennis Perzanowski, and Alan Schultz. Using a hand-drawn sketch to control a team of robots. *Autonomous Robots*, 22:399–410, 2007. 20

[277] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022. 20

[278] Jianzong Wu, Xiangtai Li, Shilin Xu Haobo Yuan, Henghui Ding, Yibo Yang, Xia Li, Jiangning Zhang, Yunhai Tong, Xudong Jiang, Bernard Ghanem, et al. Towards open vocabulary learning: A survey. *arXiv preprint arXiv:2306.15880*, 2023. 20

[279] Chenhang Cui, Yiyang Zhou, Xinyu Yang, Shirley Wu, Linjun Zhang, James Zou, and Huaxiu Yao. Holistic analysis of hallucination in gpt-4v (ision): Bias and interference challenges. *arXiv preprint arXiv:2311.03287*, 2023. 20

[280] Jensen Gao, Bidipta Sarkar, Fei Xia, Ted Xiao, Jiajun Wu, Brian Ichter, Anirudha Majumdar, and Dorsa Sadigh. Physically grounded vision-language models for robotic manipulation. In *arxiv*, 2023. 21, 23, 26

[281] Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, Sergey Levine, and Chelsea Finn. Robonet: Large-scale multi-robot learning, 2020. 21

[282] Frederik Ebert, Yanlai Yang, Karl Schmeckpeper, Bernadette Bucher, Georgios Georgakis, Kostas Daniilidis, Chelsea Finn, and Sergey Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets, 2021. 21

[283] Homer Walke, Kevin Black, Abraham Lee, Moo Jin Kim, Max Du, Chongyi Zheng, Tony Zhao, Philippe Hansen-Estruch, Quan Vuong, Andre He, et al. Bridgedata v2: A dataset for robot learning at scale. *arXiv preprint arXiv:2308.12952*, 2023. 21, 23

[284] Kenneth Shaw, Ananye Agarwal, and Deepak Pathak. Leap hand: Low-cost, efficient, and anthropomorphic hand for robot learning. *arXiv preprint arXiv:2309.06440*, 2023. 21, 28

[285] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017. 21

[286] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9068–9079, 2018. 21

[287] Manolis Savva*, Abhishek Kadian*, Oleksandr Maksymets*, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 21

[288] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, et al. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017. 22

[289] So Yeon Min, Devendra Singh Chaplot, Pradeep Ravikumar, Yonatan Bisk, and Ruslan Salakhutdinov. Film: Following instructions in language with modular methods, 2022. 22

[290] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles, 2017. 22

[291] Google DeepMind. Mujoco 3.0. https://github.com/google-deepmind/mujoco/releases/tag/3.0.0, 2023. Accessed: [Insert date of access]. 22

[292] S. Dass, J. Yapeter, J. Zhang, J. Zhang, K. Pertsch, S. Nikolaidis, and J. J. Lim. Clvr jaco play dataset. https://github.com/clvrai/clvr_jaco_play_dataset, 2023. 23

[293] Jianlan Luo, Charles Xu, Xinyang Geng, Gilbert Feng, Kuan Fang, Liam Tan, Stefan Schaal, and Sergey Levine. Multi-stage cable routing through hierarchical imitation learning, 2023. 23

[294] Jyothish Pari, Nur Muhammad Shafiullah, Sridhar Pandian Arunachalam, and Lerrel Pinto. The surprising effectiveness of representation learning for visual imitation. *arXiv preprint arXiv:2112.01511*, 2021. 23

[295] L. Y. Chen, S. Adebola, and K. Goldberg. Berkeley ur5 demonstration dataset. https://sites.google.com/view/berkeley-ur5/home. Accessed: [Insert Date Here]. 23

[296] Erick Rosete-Beas, Oier Mees, Gabriel Kalweit, Joschka Boedecker, and Wolfram Burgard. Latent plans for task agnostic offline reinforcement learning. In *Proceedings of the 6th Conference on Robot Learning (CoRL)*, 2022. 23

[297] Xufeng Zhao, Mengdi Li, Cornelius Weber, Muhammad Burhan Hafez, and Stefan Wermter. Chat with the environment: Interactive multimodal perception using large language models, 2023. 23

[298] Coppelia Robotics. Coppeliasim. https://www.coppeliarobotics.com/. Accessed: [Insert Date Here]. 23

[299] E. Coumans and Y. Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. 23

[300] Siyuan Huang, Zhengkai Jiang, Hao Dong, Yu Qiao, Peng Gao, and Hongsheng Li. Instruct2act: Mapping multi-modality instructions to robotic actions with large language model, 2023. 23

[301] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11097–11107, 2020. 23

[302] Kaichun Mo, Leonidas Guibas, Mustafa Mukadam, Abhinav Gupta, and Shubham Tulsiani. Where2act: From pixels to actions for articulated 3d objects, 2021. 23

[303] Alex X. Lee, Coline Devin, Yuxiang Zhou, Thomas Lampe, Konstantinos Bousmalis, Jost Tobias Springenberg, Arunkumar Byravan, Abbas Abdolmaleki, Nimrod Gileadi, David Khosid, Claudio Fantacci, Jose Enrique Chen, Akhil Raju, Rae Jeong, Michael Neunert, Antoine Laurens, Stefano Saliceti, Federico Casarini, Martin Riedmiller, Raia Hadsell, and Francesco Nori. Beyond pick-and-place: Tackling robotic stacking of diverse shapes, 2021. 24

[304] Andrew Szot, Max Schwarzer, Harsh Agrawal, Bogdan Mazoure, Walter Talbott, Katherine Metcalf, Natalie Mackraz, Devon Hjelm, and Alexander Toshev. Large language models as generalizable policies for embodied tasks, 2023. 24

[305] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. 24

[306] Andy Zeng, Pete Florence, Jonathan Tompson, Stefan Welker, Jonathan Chien, Maria Attarian, Travis Armstrong, Ivan Krasin, Dan Duong, Vikas Sindhwani, and Johnny Lee. Transporter networks: Rearranging the visual world for robotic manipulation. *Conference on Robot Learning (CoRL)*, 2020. 24, 25

[307] Yecheng Jason Ma, William Liang, Vaidehi Som, Vikash Kumar, Amy Zhang, Osbert Bastani, and Dinesh Jayaraman. Liv: Language-image representations and rewards for robotic control, 2023. 24

[308] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Avnish Narayan, Hayden Shively, Adithya Bellathur, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning, 2021. 24, 25

[309] Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. Tidybot: Personalized robot assistance with large language models. *arXiv preprint arXiv:2305.05658*, 2023. 24

[310] Yan Ding, Xiaohan Zhang, Chris Paxton, and Shiqi Zhang. Task and motion planning with large language models for object rearrangement. *arXiv preprint arXiv:2303.06247*, 2023. 24

[311] N. Koenig and A. Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, volume 3, pages 2149–2154 vol.3, 2004. 24

[312] Karmesh Yadav, Ram Ramrakhya, Santhosh Kumar Ramakrishnan, Theo Gervet, John Turner, Aaron Gokaslan, Noah Maestre, Angel Xuan Chang, Dhruv Batra, Manolis Savva, Alexander William Clegg, and Devendra Singh Chaplot. Habitat-matterport 3d semantics dataset, 2023. 24, 25

[313] Théophane Weber, Sébastien Racaniere, David P Reichert, Lars Buesing, Arthur Guez, Danilo Jimenez Rezende, Adria Puigdomenech Badia, Oriol Vinyals, Nicolas Heess, Yujia Li, et al. Imagination-augmented agents for deep reinforcement learning. *arXiv preprint arXiv:1707.06203*, 2017. 25

[314] Maxime Chevalier-Boisvert, Dzmitry Bahdanau, Salem Lahlou, Lucas Willems, Chitwan Saharia, Thien Huu Nguyen, and Yoshua Bengio. Babyai: A platform to study the sample efficiency of grounded language learning. *arXiv preprint arXiv:1810.08272*, 2018. 25

[315] Karl Cobbe, Chris Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*, pages 2048–2056. PMLR, 2020. 25

[316] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013. 25

[317] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018. 25

[318] Wenlong Huang, Fei Xia, Dhruv Shah, Danny Driess, Andy Zeng, Yao Lu, Pete Florence, Igor Mordatch, Sergey Levine, Karol Hausman, and Brian Ichter. Grounded decoding: Guiding text generation with grounded models for robot control. *ArXiv*, abs/2303.00855, 2023. 25

[319] M. Chevalier-Boisvert, L. Willems, and S. Pal. Minimalistic gridworld environment for gymnasium. https://github.com/pierg/environments-rl, 2018. 25

[320] Yuanpei Chen, Tianhao Wu, Shengjie Wang, Xidong Feng, Jiechuan Jiang, Zongqing Lu, Stephen McAleer, Hao Dong, Song-Chun Zhu, and Yaodong Yang. Towards human-level bimanual dexterous manipulation with reinforcement learning. *Advances in Neural Information Processing Systems*, 35:5150–5163, 2022. 25

[321] Taylor Howell, Nimrod Gileadi, Saran Tunyasuvunakool, Kevin Zakka, Tom Erez, and Yuval Tassa. Predictive sampling: Real-time behaviour synthesis with mujoco, 2022. 25

[322] Matthew Chignoli, Donghyun Kim, Elijah Stanger-Jones, and Sangbae Kim. The mit humanoid robot: Design, motion planning, and control for acrobatic behaviors. In *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, pages 1–8. IEEE, 2021. 22

[323] Weiye Zhao, Tairan He, and Changliu Liu. Model-free safe control for zero-violation reinforcement learning. In *5th Annual Conference on Robot Learning*, 2021. 26

[324] Sylvia Herbert, Jason J. Choi, Suvansh Sanjeev, Marsalis Gibson, Koushil Sreenath, and Claire J. Tomlin. Scalable learning of safety guarantees for autonomous systems using hamilton-jacobi reachability, 2021. 26, 27

[325] Ran Tian, Liting Sun, Andrea Bajcsy, Masayoshi Tomizuka, and Anca D Dragan. Safety assurances for human-robot interaction via confidence-aware game-theoretic human models. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 11229–11235. IEEE, 2022. 27

[326] S.H. Cheong, J.H. Lee, and C.H. Kim. A new concept of safety affordance map for robots object manipulation. In *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 565–570, 2018. 27

[327] Ziyi Yang, Shreyas Sundara Raman, Ankit Shah, and Stefanie Tellex. Plug in the safety chip: Enforcing constraints for LLM-driven robot agents. In *2nd Workshop on Language and Robot Learning: Language as Grounding*, 2023. 27

[328] Olaf Sporns and Richard F Betzel. Modular brain networks. *Annual review of psychology*, 67:613–640, 2016. 27

[329] David Meunier, Renaud Lambiotte, and Edward T Bullmore. Modular and hierarchically modular organization of brain networks. *Frontiers in neuroscience*, 4:200, 2010. 27

[330] Montserrat Gonzalez Arenas, Ted Xiao, Sumeet Singh, Vidhi Jain, Allen Z. Ren, Quan Vuong, Jake Varley, Alexander Herzog, Isabel Leal, Sean Kirmani, Dorsa Sadigh, Vikas Sindhwani, Kanishka Rao, Jacky Liang, and Andy Zeng. How to prompt your robot: A promptbook for manipulation skills with code as policies. In *2nd Workshop on Language and Robot Learning: Language as Grounding*, 2023. 28

[331] Zengyi Qin, Kuan Fang, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. Keto: Learning keypoint representations for tool manipulation, 2019. 28

[332] Dylan Turpin, Liquang Wang, Stavros Tsogkas, Sven J. Dickinson, and Animesh Garg. Gift: Generalizable interaction-aware functional tool affordances without labels. *ArXiv*, abs/2106.14973, 2021.

[333] Carl Qi, Sarthak Shetty, Xingyu Lin, and David Held. Learning generalizable tool-use skills through trajectory generation. *ArXiv*, abs/2310.00156, 2023. 28

[334] Shadow Robot Company. Dexterous hand series. https://www.shadowrobot.com/dexterous-hand-series/, 2023. Accessed: 2023-12-10. 28

[335] Siyuan Dong, Wenzhen Yuan, and Edward H. Adelson. Improved gelsight tactile sensor for measuring geometry and slip. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, September 2017. 28

[336] Zilin Si, Tianhong Catherine Yu, Katrene Morozov, James McCann, and Wenzhen Yuan. Robotsweater: Scalable, generalizable, and customizable machine-knitted tactile skins for robots. *arXiv preprint arXiv:2303.02858*, 2023. 28

[337] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware, 2023. 28

[338] Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, and Kentaro Inui. Realtime qa: What's the answer right now?, 2022. 29

[339] Sanket Vaibhav Mehta, Jai Gupta, Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Jinfeng Rao, Marc Najork, Emma Strubell, and Donald Metzler. Dsi++: Updating transformer memory with new documents. *ArXiv*, abs/2212.09744, 2022.

[340] Sanket Vaibhav Mehta, Darshan Patil, Sarath Chandar, and Emma Strubell. An empirical investigation of the role of pre-training in lifelong learning. *Journal of Machine Learning Research*, 24(214):1–50, 2023.

[341] James Seale Smith, Paola Cascante-Bonilla, Assaf Arbelle, Donghyun Kim, Rameswar Panda, David Cox, Diyi Yang, Zsolt Kira, Rogerio Feris, and Leonid Karlinsky. Construct-vl: Data-free continual structured vl concepts learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14994–15004, 2023. 29

[342] Timothée Lesort, Vincenzo Lomonaco, Andrei Stoian, Davide Maltoni, David Filliat, and Natalia Díaz-Rodríguez. Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges. *Information Fusion*, 58:52–68, June 2020. 29

[343] Guilherme Maeda, Marco Ewerton, Takayuki Osa, Baptiste Busch, and Jan Peters. Active incremental learning of robot movement primitives. In Sergey Levine, Vincent Vanhoucke, and Ken Goldberg, editors, *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pages 37–46. PMLR, 13–15 Nov 2017. 29

[344] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. In *Robotics: Science and Systems*, 2021. 29

[345] Joey Hejna and Dorsa Sadigh. Inverse preference learning: Preference-based rl without a reward function, 2023. 29

[346] Zihao Li, Zhuoran Yang, and Mengdi Wang. Reinforcement learning with human feedback: Learning dynamic choices via pessimism, 2023. 29

[347] Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2022. 29

[348] OpenAI, Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, Jonas Schneider, Nikolas Tezak, Jerry Tworek, Peter Welinder, Lilian Weng, Qiming Yuan, Wojciech Zaremba, and Lei Zhang. Solving rubik's cube with a robot hand, 2019. 29

[349] Viraj Mehta, Vikramjeet Das, Ojash Neopane, Yijia Dai, Ilija Bogunovic, Jeff Schneider, and Willie Neiswanger. Sample efficient reinforcement learning from human feedback via active exploration, 2023. 29

[350] Gaon An, Junhyeok Lee, Xingdong Zuo, Norio Kosaka, Kyung-Min Kim, and Hyun Oh Song. Direct preference-based policy optimization without reward modeling, 2023. 29

[351] Anurag Ajay, Seungwook Han, Yilun Du, Shuang Li, Abhi Gupta, Tommi Jaakkola, Josh Tenenbaum, Leslie Kaelbling, Akash Srivastava, and Pulkit Agrawal. Compositional foundation models for hierarchical planning, 2023. 29

[352] Kevin Frans, Jonathan Ho, Xi Chen, Pieter Abbeel, and John Schulman. Meta learning shared hierarchies, 2017.

[353] Suraj Nair and Chelsea Finn. Hierarchical foresight: Self-supervised learning of long-horizon tasks via visual subgoal generation. In *International Conference on Learning Representations*, 2020. 29

[354] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition, 2019. 29