arXiv:2103.04918v8 [cs.AI] 5 Jan 2022

A Survey of Embodied AI: From Simulators to Research Tasks

Jiafei Duan, Samson Yu, Hui Li Tan, Hongyuan Zhu, Cheston Tan

***Abstract*—There has been an emerging paradigm shift from the era of "internet AI" to "embodied AI", where AI algorithms and agents no longer learn from datasets of images, videos or text curated primarily from the internet. Instead, they learn through interactions with their environments from an egocentric perception similar to humans. Consequently, there has been substantial growth in the demand for embodied AI simulators to support various embodied AI research tasks. This growing interest in embodied AI is beneficial to the greater pursuit of Artificial General Intelligence (AGI), but there has not been a contemporary and comprehensive survey of this field. This paper aims to provide an encyclopedic survey for the field of embodied AI, from its simulators to its research. By evaluating nine current embodied AI simulators with our proposed seven features, this paper aims to understand the simulators in their provision for use in embodied AI research and their limitations. Lastly, this paper surveys the three main research tasks in embodied AI – visual exploration, visual navigation and embodied question answering (QA), covering the state-of-the-art approaches, evaluation metrics and datasets. Finally, with the new insights revealed through surveying the field, the paper will provide suggestions for simulator-for-task selections and recommendations for the future directions of the field.**

***Index Terms*—Embodied AI, Computer Vision, 3D Simulators.**

I. INTRODUCTION

RECENT advances in deep learning, reinforcement learning, computer graphics and robotics have garnered growing interest in developing general-purpose AI systems. As a result, there has been a shift from "internet AI" that focuses on learning from datasets of images, videos and text curated from the internet, towards "embodied AI" which enables artificial agents to learn through interactions with their surrounding environments. Embodied AI is the belief that true intelligence can emerge from the interactions of an agent with its environment [1]. But for now, embodied AI is about incorporating traditional intelligence concepts from vision, language, and reasoning into an artificial embodiment to help solve AI problems in a virtual environment.

The growing interest in embodied AI has led to significant progress in embodied AI simulators that aim to faithfully replicate the physical world. These simulated worlds serve as virtual testbeds to train and test embodied AI frameworks before deploying them into the real world. These embodied AI simulators also facilitate the collection of task-based dataset [2], [3] which are tedious to collect in real-world as it requires an extensive amount of manual labor to replicate the same setting as in the virtual world. While there have been several survey papers in the field of embodied AI [4]–[6], they are mostly outdated as they were published before the modern deep learning era, which started around 2009 [7]–[10]. To the best of our knowledge, there is only one survey paper on the evaluating embodied navigation [11] .

To address the scarcity of contemporary comprehensive survey papers on this emerging field of embodied AI, we propose this survey paper on the field of embodied AI, from its simulators to research tasks. This paper covers the following nine embodied AI simulators that were developed over the past four years: DeepMind Lab [12], AI2-THOR [13], CHALET [14], VirtualHome [15], VRKitchen [16], Habitat-Sim [17], iGibson [18], SAPIEN [19], and ThreeDWorld [20]. The chosen simulators are designed for general-purpose intelligence tasks, unlike game simulators [21] which are only used for training reinforcement learning agents. These embodied AI simulators provide realistic representations of the real world in computer simulations, mainly taking the configurations of rooms or apartments that provide some forms of constraint to the environment. The majority of these simulators minimally comprise a physics engine, Python API, and artificial agent that can be controlled or manipulated within the environment.

Embodied AI simulators have given rise to a series of potential embodied AI research tasks, such as *visual exploration*, *visual navigation* and *embodied QA*. We will focus on these three tasks since most existing papers [11], [22], [23] in embodied AI either focus on these tasks or make use of modules introduced for these tasks to build models for more complex tasks like audio-visual navigation. These three tasks are also connected in increasing complexity. Visual exploration is a very useful component in visual navigation [22], [24] and used for realistic situations [25], [26], while embodied QA further involves complex QA capabilities that builds on top of vision-and-language navigation. Since language is a common modality and visual QA is a popular task in AI, embodied QA is a natural direction for embodied AI. These three tasks

This work was supported by the Agency for Science, Technology and Research (A*STAR), Singapore under its AME Programmatic Funding Scheme (Award #A18A2b0046) and the National Research Foundation, Singapore under its NRFISF Joint Call (Award NRF2015-NRF-ISF001-2541).

J. Duan was with the Nanyang Technological University of Singapore, School of Electrical and Electronics Engineering, Singapore 639798, Singapore (e-mail: duan0038@e.ntu.edu.sg).

S. Yu was with the Singapore University of Technology and Design, Singapore 487372, Singapore (e-mail: samson_yu@sutd.edu.sg).

H.L. Tan, H. Zhu, and C. Tan are with the Institute for Infocomm Research, A*STAR, Singapore 138632, Singapore (e-mail: {duan_jiafei, hltan, zhuh, cheston-tan}@i2r.a-star.edu.sg).

Manuscript accepted December 4, 2021, IEEE-TETCI. *©2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.*

discussed in this paper have been implemented in at least one of the nine proposed embodied AI simulators. However, Sim2Real [27]–[29] and robotics in the physical world will not be covered in this paper.

These simulators are selected based on the embodied AI simulators from the Embodied AI Challenge in the annual Embodied AI workshop [30] at *Conference on Computer Vision and Pattern Recognition* (CVPR). The research tasks are then sourced from direct citations of these simulators.

To this end, we will provide a contemporary and comprehensive survey of embodied AI simulators and research through reviewing the development of the field from its simulator to research. In section I, this paper outlines the overview structure of this survey paper. In section II, this paper benchmarks nine embodied AI simulators to understand their provision for realism, scalability, interactivity and hence use in embodied AI research. Finally, based upon the simulators, in section III, this paper surveys three main research tasks in embodied AI - visual exploration, visual navigation and embodied question answering (QA), covering the state-of-the-art approaches, evaluation, and datasets. Lastly, this paper will establish interconnections between the simulators, datasets and research tasks and existing challenges in embodied AI simulators and research in section IV. This survey paper provides a comprehensive look into the emerging field of embodied AI and further unveils new insights and challenges of the field. Furthermore, through this paper, we seek to avail AI researchers in selecting the ideal embodied AI simulators for their research tasks of interest.

II. Simulators for Embodied AI

.

In this section, the backgrounds of the embodied AI simulators will be presented in the supplementary material, and the features of the embodied AI simulators will be compared and discussed in section II-A.

A. Embodied AI Simulators

This section presents the backgrounds of the nine embodied AI simulators: DeepMind Lab, AI2-THOR, SAPIEN, VirtualHome, VRKitchen, ThreeDWorld, CHALET, iGibson, and Habitat-Sim. Readers can refer to the supplementary material for more details on the respective simulators. In this section, the paper will comprehensively compare the nine embodied AI simulators based on seven technical features. Referencing [13], [20], [31], these seven technical features are selected as the primary features to evaluate the embodied AI simulator as they cover the essential aspects required to replicate the environment accurately, interactions and state of the physical world, hence providing suitable testbeds for testing intelligence with embodiment. Referring to Table I, the seven features are: Environment, Physics, Object Type, Object Property, Controller, Action, and Multi-Agent.

Environment: There are two main methods of constructing the embodied AI simulator environment: game-based scene construction (G) and world-based scene construction (W). Referring to Fig. 1, the game-based scenes are constructed from 3D assets, while world-based scenes are constructed from real-world scans of the objects and the environment. A 3D environment constructed entirely out of 3D assets often has built-in physics features and object classes that are well-segmented when compared to a 3D mesh of an environment made from real-world scanning. The clear object segmentation for the 3D assets makes it easy to model them as articulated objects with movable joints, such as the 3D models provided in PartNet [32]. In contrast, the real-world scans of environments and objects provide higher fidelity and more accurate representation of the real-world, facilitating better transfer of agent performance from simulation to the real world. As observed in Table I, most simulators other than Habitat-Sim and iGibson have game-based scenes, since significantly more resources are required for world-based scene construction.

Physics: A simulator has to construct not only realistic environments but also realistic interactions between agents and objects or objects and objects that model real-world physics properties. We study the simulators' physics features, which we broadly classify into basic physics features (B) and advanced physics features (A). Referring to Fig. 2, basic physics features include collision, rigid-body dynamics, and gravity modelling while advanced physics features include cloth, fluid, and soft-body physics. As most embodied AI simulators construct game-based scenes with in-built physics engines, they are equipped with the basic physics features. On the other hand, for simulators like ThreeDWorld, where the goal is to understand how the complex physics environment can shape the decisions of the artificial agent in the environment, they are equipped with more advanced physics capabilities. For simulators that focus on interactive navigation-based tasks, basic physics features are generally sufficient.

Object Type: As shown in Fig. 3, there are two main sources for objects that are used to create the simulators. The first type is the dataset driven environment, where the objects are mainly from existing object datasets such as the SUNCG [33] dataset, the Matterport3D dataset [34] and the Gibson dataset [35]. The second type is the asset driven environment, where the objects are from the net such as the Unity 3D game asset store. A difference between the two sources is the sustainability of the object dataset. The dataset driven objects are more costly to collect than the asset driven objects, as anyone can contribute to the 3D object models online. However, it is harder to ensure the quality of the 3D object models in the asset driven objects than in the dataset driven objects. Based on our review, the game-based embodied AI simulators are more likely to obtain their object datasets from asset stores, whereas the world-based simulators tend to import their object datasets from existing 3D object datasets.

Object Property: Some simulators only enable objects with basic interactivity such as collision. Advanced simulators enable objects with more fine-grained interactivity such as multiple-state changes. For instance, when an apple is sliced, it will undergo a state change into apple slices. Hence, we categorize these different levels of object interaction into simulators with interact-able objects (I) and multiple-state objects (M). Referring to Table I, a few simulators, such as AI2-THOR and VRKitchen, enable multiple state changes,

TABLE I
SUMMARY OF EMBODIED AI SIMULATORS. ENVIRONMENT: GAME-BASED SCENE CONSTRUCTION (G) AND WORLD-BASED SCENE CONSTRUCTION (W). PHYSICS: BASIC PHYSICS FEATURES (B) AND ADVANCED PHYSICS FEATURES (A). OBJECT TYPE: DATASET DRIVEN ENVIRONMENTS (D) AND OBJECT ASSETS DRIVEN ENVIRONMENTS (O). OBJECT PROPERTY: INTERACT-ABLE OBJECTS (I) AND MULTI-STATE OBJECTS (M). CONTROLLER: DIRECT PYTHON API CONTROLLER (P), VIRTUAL ROBOT CONTROLLER(R) AND VIRTUAL REALITY CONTROLLER (V). ACTION: NAVIGATION (N), ATOMIC ACTION (A) AND HUMAN-COMPUTER INTERACTION (H). MULTI-AGENT: AVATAR-BASED (AT) AND USER-BASED (U). THE SEVEN FEATURES CAN BE FURTHER GROUPED UNDER THREE SECONDARY EVALUATION FEATURES; REALISM, SCALABILITY AND INTERACTIVITY.

Year	Embodied AI Simulator	Environment (Realism)	Physics (Realism)	Object Type (Scalability)	Object Property (Interactivity)	Controller (Interactivity)	Action (Interactivity)	Multi-agent (Interactivity)
2016	DeepMind Lab	G	-	-	-	P, R	N	-
2017	AI2-THOR	G	B	O	I, M	P, R	A, N	U
2018	CHALET	G	B	O	I, M	P	A, N	-
2018	VirtualHome	G	-	O	I, M	R	A, N	-
2019	VRKitchen	G	B	O	I, M	P, V	A, N, H	-
2019	Habitat-Sim	W	-	D	-	-	N	-
2019	iGibson	W	B	D	I	P, R	A, N	U
2020	SAPIEN	G	B	D	I, M	P, R	A, N	-
2020	ThreeDWorld	G	B, A	O	I	P, R, V	A, N, H	AT

TABLE II
COMPARISON OF EMBODIED AI SIMULATORS IN TERMS OF ENVIRONMENT CONFIGURATION, SIMULATION ENGINE, TECHNICAL SPECIFICATION, AND RENDERING PERFORMANCE.

Embodied AI Simulator	Environment Configuration	Simulation Engine	Technical Specification	Rendering Performance
DeepMind Lab	Customized environment	Quake II Arena Engine	6-core Intel Xeon CPU and an NVIDIA Quadro K600 GPU	158 fps/thread
AI2-THOR	120 rooms, 4 categories	Unity 3D Engine	Intel(R) Xeon(R) CPU E5-2620 v4 and NVIDIA Titan X	240 fps/thread
CHALET	58 rooms, 10 houses	Unity 3D Engine	-	-
VirtualHome	6 apartments with multiple jointed rooms	Unity 3D Engine	-	Customized frame rate
VRKitchen	16 kitchens	Unreal Engine 4	Intel(R) Core(TM) i7-7700K processor and NVIDIA Titan X	15 fps/thread
Habitat-Sim	Mutiple datasets	-	Xeon E5-2690 v4 CPU and Nvidia Titan Xp GPU	10,000 fps/thread
iGibson	Gibson V1	-	Modern GPU	1000 fps/thread
SAPIEN	Customized environment	PhysX Physical engine and ROS	Intel i7-8750 CPU and an Nvidia GeForce RTX 2070 GPU	700 fps/thread
ThreeDWorld	Customized environment	Unity 3D Engine	Intel i7-7700K GPU: NVIDIA GeForce GTX 1080	168 fps/thread

providing a platform for understanding how objects will react and change their states when acted upon in the real world.

Controller: Referring to Fig. 4, there are different types of controller interface between the user and simulator, from direct Python API controller (P) and virtual robot controller(R) to virtual reality controller (V). Robotics embodiment allows for virtual interaction of existing real-world robots such as Universal Robot 5 (UR5) and TurtleBot V2, and can be controlled directly using a ROS interface. The virtual reality controller interfaces provide more immersive human-computer interaction and facilitate deployment using their real-world counterparts. For instance, simulators such as iGibson and AI2-THOR, which are primarily designed for visual navigation, are also equipped with virtual robot controllerfor ease of deployment in their real-world counterparts such as iGibson's Castro [36] and RoboTHOR [37] respectively.

Action: There are differences in the complexity of an artificial agent's action capabilities in the embodied AI simulator, ranging from being only able to perform primary navigation manoeuvers to higher-level human-computer actions via virtual reality interfaces. This paper classifies them into three tiers of robotics manipulation: navigation (N), atomic action (A) and human-computer interaction (H). Navigation is the lowest tier and is a common feature in all embodied AI simulators [38]. It is defined by the agent's capability of navigating around its virtual environment. Atomic action provides the artificial agent with a means of performing basic discrete manipulation to an object of interest and is found in most embodied AI simulators. Human-computer interaction is the result of the virtual reality controller as it enables humans to control virtual agents to learn and interact with the simulated world in real time [16]. Most of the larger-scale navigation-based simulators, such as AI2-THOR, iGibson and Habitat-Sim, tend to have navigation, atomic action and ROS [13], [17], [35] which enable them to provide better control and manipulation of objects in the environment while performing tasks such as Point Navigation or Object Navigation. On the other hand, simulators such as ThreeDWorld and VRKitchen [16], [20] fall under the human-computer interaction category as they are constructed to provide a highly realistic physics-based simulation and multiple state changes. This is only possible with human-computer interaction as human-level dexterity is needed when interacting with these virtual objects.

Multi-agent: Referring to Table I, only a few simulators, such as AI2-THOR, iGibson and ThreeDWorld, are equipped with multi-agent setup, as current research involving multi-agent reinforcement learning is scarce. In general, the simulators need to be rich in object content before there is any practical value of constructing such multi-agent features used for both adversarial and collaborative training [39], [40] of artificial agents. As a result of this lack of multi-agent supported simulators, there have been fewer research tasks that utilize the multi-agent feature in these embodied AI simulators.

For multi-agent reinforcement learning based training, they are still currently being done in OpenAI Gym environments [41] . There are two distinct multi-agent settings. The first is the avatar-based (AT) multi-agents in ThreeDWorld [20] that allows for interaction between artificial agents and simulation avatars. The second is the user-based (U) multi-agents in AI2-THOR [13] which can take on the role of a dual learning network and learn from interacting with other artificial agents in the simulation to achieve a common task [42].

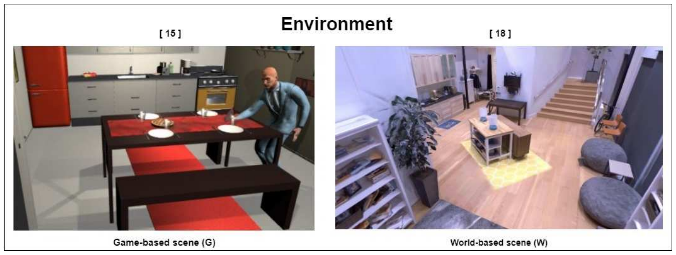

Fig. 1. Comparison between game-based scene (G) and world-based scene (W). The game-based scene (G) focuses on environment that are constructed from 3D object assets, while the world-based scene (W) are constructed based off real-world scans of the environment.

B. Comparison of Embodied AI Simulators

Constructed on the seven features and a study from the Allen Institute of Artificial Intelligence [31] on embodied AI, we propose a secondary set of evaluation features for the simulators. It comprises of three key features: *realism*, *scalability* and *interactivity* as shown in Table I. The realism of the 3D environments can be attributed to the *environment* and *physics* of the simulators. The environment models the real world's physical appearance while the physics models the complex physical properties within the real world. Scalability of the 3D environments can be attributed to the *object type*. The expansion can be done via collecting more 3D scans of the real world for the dataset driven objects or purchasing more 3D assets for the asset driven objects. *Interactivity* is attributed to *object property*, *controller*, *action* and *multi-agent*.

Based on the secondary evaluation features of embodied AI simulators, the seven primary features from the Table I and the Fig. 6, simulators which possess all of the above three secondary features (e.g. AI2-THOR, iGibson and Habitat-Sim) are more well-received and widely used for a diverse range of embodied AI research tasks. Furthermore, a comprehensive quantitative comparison is made for all the embodied AI simulators to compare the environment configuration and the technical performance of each simulator. The **environment configuration** feature is very much dependent on the applications suggested by the creators of the simulators, while other features like **technical specification** and **rendering performance** are largely due to the **simulation engine** used for its creation. AI2-THOR has the largest environment configurations compared to the other simulators, while Habitat-Sim and iGibson are the top two performers in graphic rendering performance. This benchmark of quantitative performance shown in Table II further demonstrates the superiority and complexity of these three embodied AI simulators. These comparisons of the embodied AI simulators further have reinforced the importance of the seven primary evaluation metrics and the three secondary evaluations that the paper has established to help select the ideal simulator for the research task.

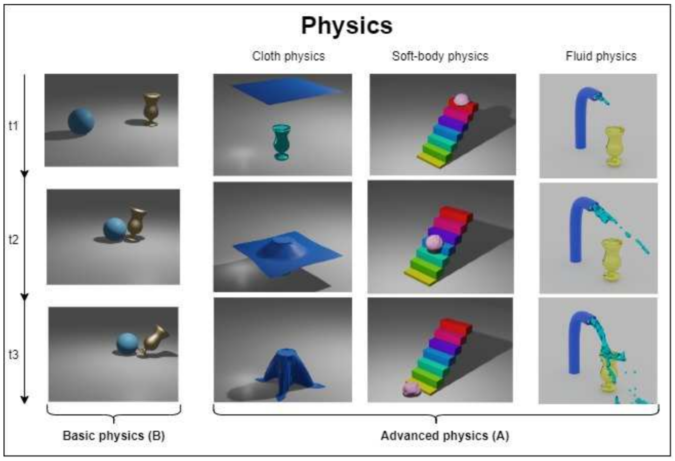

Fig. 2. Comparison between basics physics features such as rigid-body and collision (B) and advanced physics features (A) which includes cloth, soft-body, and fluid physics.

III. Research in Embodied AI

In this section, we discuss the various embodied AI research tasks that depend on the nine embodied AI simulators surveyed in the previous section. There are multiple motivations for the recent increase in embodied AI research. From a cognitive science and psychology perspective, the embodiment hypothesis [1] suggests that intelligence arises from interactions with an environment and as a result of sensorimotor activity [66]. Intuitively, humans do not learn solely through the "internet AI" paradigm where most experiences are randomized and passive (i.e. externally curated). Humans also learn through active perception, movement, interaction and communication. From an AI perspective, current research tasks in embodied AI allows for greater generalization to unseen environments [44] for robotic functions like mapping and navigation and greater robustness to sensor noise as compared to classical methods due to the learning involved. Embodied AI also enables flexibility and possibly greater performance since various modalities like depth, language [59] and audio [67] can be easily integrated through learning-based approaches.

The three main types of embodied AI research tasks are *visual exploration*, *visual navigation* and *embodied QA*. We will focus on these three tasks since most existing papers in embodied AI either focus on these tasks or make use of modules introduced for these tasks to build models for more complex tasks like audio-visual navigation. The tasks increase in complexity as it advances from exploration to QA. We will start with the visual exploration before moving to visual navigation and finally embodied QA. Each of these tasks makes up the foundation for the next task(s), forming a pyramid structure of embodied AI research tasks as shown in Fig. 5, further suggesting a natural direction for embodied AI. We will highlight important aspects for each task, starting with the summary, the methodologies, evaluation metrics, to the datasets. These task details are found in Table III.

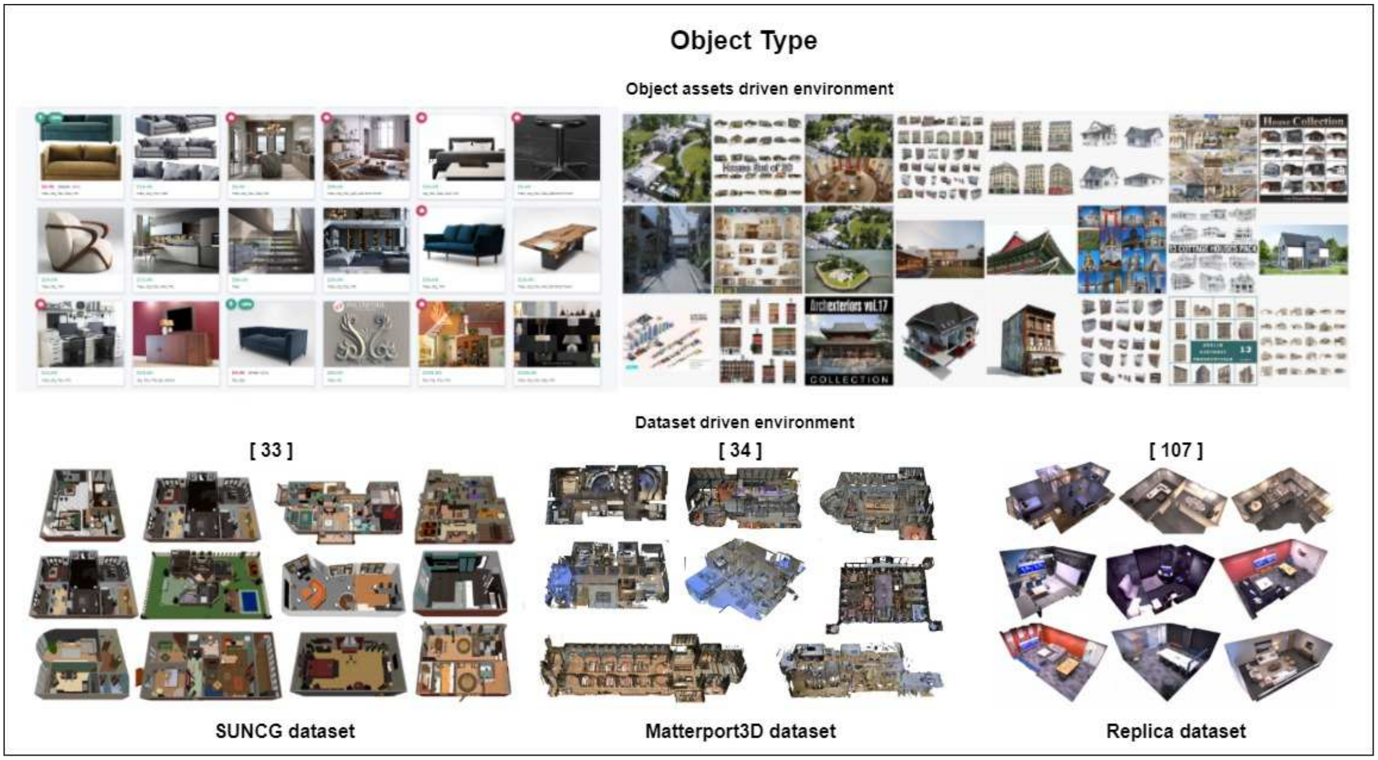

Fig. 3. Comparison between dataset driven environment (D) which are constructed from 3D objects datasets and object assets driven environment (O) are constructed based 3D objects obtain from the assets market.

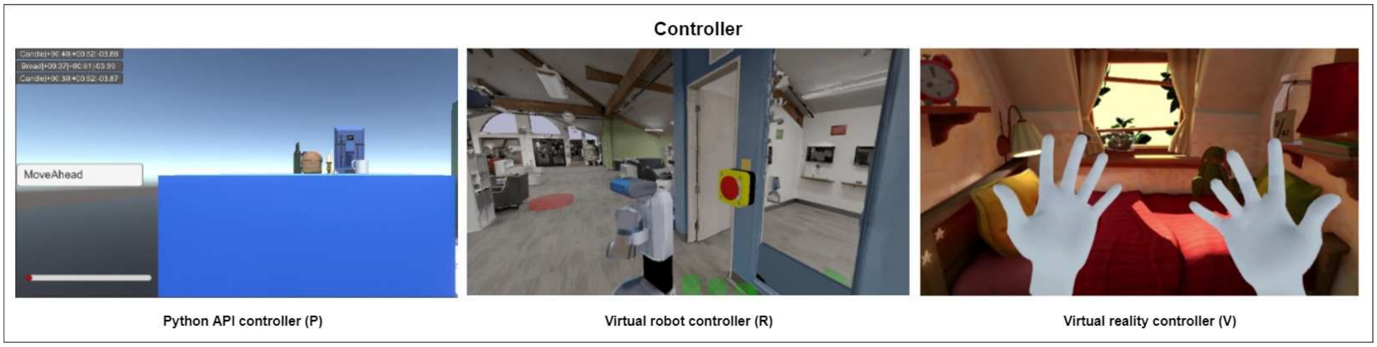
Controller
MoveAhead
Python API controller (P)
Virtual robot controller (R)
Virtual reality controller (V)

Fig. 4. Comparison between direct Python API controller (P), robotics embodiment (R) which refers to real-world robots with a virtual replica and lastly the virtual reality controller (V).

A. Visual Exploration

In visual exploration [24], [68], an agent gathers information about a 3D environment, typically through motion and perception, to update its internal model of the environment [11], [22], which might be useful for downstream tasks like visual navigation [24], [25], [69]. The aim is to do this as efficiently as possible (e.g. with as few steps as possible). The internal model can be in forms like a topological graph map [26], semantic map [46], occupancy map [45] or spatial memory [70], [71]. These map-based architectures can capture geometry and semantics, allowing for more efficient policy learning and planning [45] as compared to reactive and recurrent neural network policies [72]. Visual exploration is usually either done before or concurrently with navigation tasks. In the first case, visual exploration builds the internal memory as priors that are useful for path-planning in downstream navigation tasks. The agent is free to explore the environment within a certain budget (e.g. limited number of steps) before the start of navigation [11]. In the latter case, the agent builds the map as it navigates an unseen test environment [48], [73], [74], which makes it more tightly integrated with the downstream task. In this section, we build upon existing visual exploration survey papers [22], [24] to include more recent works and directions.

In classical robotics, exploration is done through passive or active simultaneous localisation and mapping (SLAM) [24], [45] to build a map of the environment. This map is then used with localization and path-planning for navigation tasks. SLAM is very well-studied [75], but the purely geometric approach has room for improvements. Since they rely on sensors, they are susceptible to measurement noise [24] and would need extensive fine-tuning. On the other hand, learning-

TABLE III

Summary of embodied AI research tasks. Evaluation Metric: Amount of targets visited (ATV), downstream tasks (D), success weighted by path length (SPL), success rate (SR), path length ratio (PLR), oracle success rate (OSR), trajectory/episode length (TL / EL), distance to success / navigation error (DTS / NE / d_T), goal progress (GP / d_Δ), oracle path success rate (OPSR), smallest distance to target at any point in an episode (d_{min}), percentage of episodes agent ends navigation for answering before max episode length ($\%stop$), percentage of questions agent terminates in the room containing the target object ($\%r_T$), percentage of questions where the agent enters the room containing the target oject at least once ($\%r_e$), Intersection over Union for target object (IoU), hit accuracy based on IoU (h_T), mean rank of the ground-truth answer in QA predictions (MR) and QA accuracy (Acc).

Task	Method / Category	Publication	Year	Simulator	Dataset	Evaluation Metric
Visual Exploration	Curiosity	Chaplot et al. [43]	2020	Habitat-Sim	Matterport3D, Gibson V1	ATV
	Coverage	Chaplot et al. [44]	2020	Habitat-Sim	Matterport3D, Gibson V1	ATV, D
	Reconstruction	Ramakrishnan et al. [45]	2020	Habitat-Sim	Matterport3D, Gibson V1	ATV, D
		Ramakrishnan et al. [22]	2020	Habitat-Sim	Matterport3D	ATV, D
		Narasimhan et al. [46]	2020	Habitat-Sim	Matterport3D	ATV, D
Visual Navigation	Point Navigation	Wijmans et al. [47]	2019	Habitat-Sim	Matterport3D, Gibson V1	SPL, SR
		Georgakis et al. [48]	2019	Habitat-Sim	Matterport3D	SR, PLR
		Ye et al. [49]	2020	Habitat-Sim	Gibson V1	SPL, SR
		Chaplot et al. [44]	2020	Habitat-Sim	Matterport3D, Gibson V1	SPL, SR
		Ramakrishnan et al. [45]	2020	Habitat-Sim	Matterport3D, Gibson V1	SPL, SR
		Ramakrishnan et al. [22]	2020	Habitat-Sim	Matterport3D	SPL
		Narasimhan et al. [46]	2020	Habitat-Sim	Matterport3D	SPL, SR
		Claudia, et al. [50]	2020	iGibson	Gibson V1	SR
	Object Navigation	Wortsman et al. [51]	2019	AI2-THOR	-	SPL, SR
		Campari et al. [52]	2020	Habitat-Sim	Matterport3D	SPL, SR, DTS
		Du et al. [53]	2020	AI2-THOR	-	SPL, SR
		Chaplot et al. [54]	2020	Habitat-Sim	Matterport3D, Gibson V1	SPL, SR, DTS
		Shen et al. [55]	2020	iGibson	Gibson V1	SR
		Wahid et al. [56]	2020	-	Gibson V1	SPL, SR
	Navigation with Priors	Yang et al. [57]	2020	AI2-THOR	-	SPL, SR
	Vision-and-Language Navigation	Anderson et al. [58]	2018	-	Room-to-Room	SR, OSR, TL, NE
		Zhu et al. [59]	2020	-	Room-to-Room	SPL, SR, OSR, TL, NE
		Zhu et al. [60]	2020	-	Cooperative Vision-and-Dialog Navigation	SR, OSR, GP, OPSR
Embodied Question Answering	Question Answering	Das et al. [61]	2018	-	EQA	d_T, d_Δ, d_{min}, $\%r_T$, $\%r_e$, $\%stop$, MR
		Das et al. [62]	2018	-	EQA	d_T, d_Δ, Acc
	Multi-target Question Answering	Yu et al. [63]	2019	-	MT-EQA	d_T, d_Δ, $\%r_T$, $\%stop$, IoU, h_T, Acc
	Interactive Question Answering	Gordon et al. [64]	2018	AI2-THOR	IQUAD V1	EL, Acc
		Tan et al. [65]	2020	AI2-THOR	IQUAD V1	EL, Acc

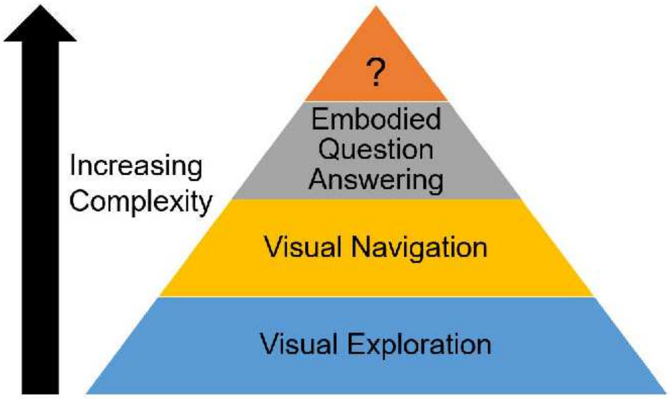

Fig. 5. A pyramid hierarchical structure of the various embodied AI research tasks with increasing complexity of tasks.

based approaches that typically use RGB and/or depth sensors are more robust to noise [24], [44]. Furthermore, learning-based approaches in visual exploration allow an artificial agent to incorporate semantic understanding (e.g. object types in the environment) [45] and generalize its knowledge of previously seen environments to help with understanding novel environments in an unsupervised manner. This reduces reliance on humans and thus improves efficiency.

Learning to create useful internal models of the environment in the form of maps can improve the agent's performance [45], whether it is done before (i.e. unspecified downstream tasks) or concurrently with downstream tasks. Intelligent exploration would also be especially useful in cases where the agent has to explore novel environments that dynamically unfold over time [76], such as rescue robots and deep-sea exploration robots.

1) Approaches: In this section, the non-baseline approaches in visual exploration are typically formalized as partially observed Markov decision processes (POMDPs) [77]. A POMDP can be represented by a 7-tuple $(S, A, T, R, \Omega, O, \gamma)$ with state space S, action space A, transition distribution T, reward function R, observation space Ω, observation distribution O and discount factor $\gamma \in [0, 1]$. In general, these approaches are viewed as a particular reward function in the POMDP [22].

Baselines. Visual exploration has a few common baselines [22]. For *random-actions* [17], the agent samples from a uniform distribution over all actions. For *forward-action*, it always chooses the forward action. For *forward-action+*, the agent chooses the forward action, but turns left if it collides. For *frontier-exploration*, it visits the edges between free and unexplored spaces iteratively using a map [24], [78].

Curiosity. In the *curiosity* approach, the agent seeks states that are difficult to predict. The prediction error is used as the reward signal for reinforcement learning [79], [80]. This focuses on intrinsic rewards and motivation rather than external rewards from the environment, which is beneficial in cases where external rewards are sparse [81]. There is usually a forward-dynamics model that minimises the loss: $L(\hat{s}_{t+1}, s_{t+1})$. In this case, $\hat{s}_{t+1}$ is the *predicted* next state if the agent takes action a_t when it is in state s_t, while s_{t+1} is the *actual* next state that the agent will end up in. Practical considerations for curiosity have been listed in recent work [79], such as using Proximal Policy Optimization (PPO) for policy optimisation. Curiosity has been used to generate more advanced maps like semantic maps in recent work [43]. Stochasticity poses a serious challenge in the curiosity approach, since the forward-dynamics model can exploit stochasticity [79] for high prediction errors (i.e. high rewards). This can arise due to factors like the "noisy-TV" problem or noise in the execution of the agent's actions [81]. One proposed solution is the use of an inverse-dynamics model [68] that estimates the action a_{t-1} taken by the agent to move from its previous state s_{t-1} to its current state s_t, which helps the agent understand what its actions can control in the environment. While this method attempts to address stochasticity due to the environment, it may be insufficient in addressing stochasticity that results from the agent's actions. One example is the agent's use of a remote controller to randomly change TV channels, allowing it to accumulate rewards without progress. To address this more challenging issue specifically, there have been a few methods proposed recently. Random Distillation Network [82] is one method that predicts the output of a randomly initialized neural network, as the answer is a deterministic function of its inputs. Another method is Exploration by Disagreement [81], where the agent is incentivised to explore the action space which has the maximum disagreement or variance between the predictions of an ensemble of forward-dynamics models. The models converges to mean, which reduces the variance of the ensemble and prevents it from getting stuck in stochasticity traps.

Coverage. In the *coverage* approach, the agent tries to maximise the amount of targets it directly observes. Typically, this would be the area seen in an environment [22], [24], [44]. Since the agent uses egocentric observations, it has to navigate based on possibly obstructive 3D structures. One recent method combines classic and learning-based methods [44]. It uses analytical path planners with a learned SLAM module that maintains a spatial map, to avoid the high sample complexities involved in training end-to-end policies. This method also includes noise models to improve physical realism for generalisability to real-world robotics. Another recent work is a scene memory transformer which uses the self-attention mechanism adapted from the Transformer model [83] over the scene memory in its policy network [72]. The scene memory embeds and stores all encountered observations, allowing for greater flexibility and scalability as compared to a map-like memory that requires inductive biases.

Reconstruction. In the *reconstruction* approach, the agent tries to recreate other views from an observed view. Past work focuses on pixel-wise reconstructions of 360 degree panoramas and CAD models [84], [85], which are usually curated datasets of human-taken photos [45]. Recent work has adapted this approach for embodied AI, which is more complex because the model has to perform scene reconstruction from the agent's egocentric observations and the control of its own sensors (i.e. active perception). In a recent work, the agent uses its egocentric RGB-D observations to reconstruct the occupancy state beyond visible regions and aggregate its predictions over time to form an accurate occupancy map [45]. The occupancy anticipation is a pixel-wise classification task where each cell in a local area of V x V cells in front of the camera is assigned probabilities of it being explored and occupied. As compared to the *coverage* approach, anticipating the occupancy state allows the agent to deal with regions that are not directly observable. Another recent work focuses on semantic reconstruction rather than pixel-wise reconstruction [22]. The agent is designed to predict whether semantic concepts like "door" are present at sampled query locations. Using a K-means approach, the true reconstruction concepts for a query location are the J nearest cluster centroids to its feature representation. The agent is rewarded if it obtains views that help it predict the true reconstruction concepts for sampled query views.

2) Evaluation Metrics: Amount of targets visited. Different types of targets are considered, such as area [44], [86] and interesting objects [72], [87]. The area visited metric has a few variants, such as the absolute coverage area in m^2 and the percentage of the area explored in the scene.

Impact on downstream tasks. Visual exploration performance can also be measured by its impact on downstream tasks like visual navigation. This evaluation metric category is more commonly seen in recent works. Examples of downstream tasks that make use of visual exploration outputs (i.e. maps) include Image Navigation [26], [73], Point Navigation [11], [44] and Object Navigation [53], [54], [56]. More details about these navigation tasks can be found in section III-B.

3) Datasets: For visual exploration, some popular datasets include Matterport3D and Gibson V1. Matterport3D and Gibson V1 are both photorealistic RGB datasets with useful information for embodied AI like depth and semantic segmentations. The Habitat-Sim simulator allows for the usage of these datasets with extra functionalities like configurable agents and multiple sensors. Gibson V1 has also been enhanced with features like interactions and realistic robot control to form iGibson. However, more recent 3D simulators like those mentioned in section II can all be used for visual exploration, since they all offer RGB observations at the very least.

B. Visual Navigation

In visual navigation, an agent navigates a 3D environment to a goal with or without external priors or natural language instruction. Many types of goals have been used for this task, such as points, objects, images [88], [89] and areas [11]. We will focus on points and objects as goals for visual navigation in this paper, as they are the most common and fundamental goals. They can be further combined with specifications like perceptual inputs and language to build towards more complex visual navigation tasks, such as *Navigation with Priors*, *Vision-and-Language Navigation* and even *Embodied QA*. Under point navigation [49], the agent is tasked to navigate to a specific point while in object navigation [38], [52], the agent is tasked to navigate to an object of a specific class.

While classic navigation approaches [90] are usually composed of hand-engineered sub-components like localization, mapping [91], path planning [92], [93] and locomotion. Visual navigation in embodied AI aims to learn these navigation systems from data, so as to reduce case-specific hand-engineering, hence easing integration with downstream tasks having superior performance with the data-driven learning methods, such as question answering [23]. There are also hybrid approaches [44] that aim to combine the best of both worlds. As previously mentioned in section II, learning-based approaches are more robust to sensor measurement noise as they use RGB and/or depth sensors and are able to incorporate semantic understanding of an environment. Furthermore, they enable an agent to generalize its knowledge of previously seen environments to help understand novel environments in an unsupervised manner, reducing human effort.

Along with the increase in research in recent years, challenges have also been organised for visual navigation in the fundamental point navigation and object navigation tasks to benchmark and accelerate progress in embodied AI [38]. The most notable challenges are the iGibson Sim2Real Challenge, Habitat Challenge [36] and RoboTHOR Challenge. For each challenge, we will describe the 2020 version of the challenges, which is the latest as of this paper. In all three challenges, the agent is limited to egocentric RGB-D observations. For the iGibson Sim2Real Challenge 2020, the specific task is point navigation. 73 high-quality Gibson 3D scenes are used for training, while the Castro scene, the reconstruction of a real world apartment, will be used for training, development and testing. There are three scenarios: when the environment is free of obstacles, contains obstacles that the agent can interact with, and/or is populated with other moving agents. For the Habitat Challenge 2020, there are both point navigation and object navigation tasks. Gibson 3D scenes with Gibson dataset splits are used for the point navigation task, while 90 Matterport3D scenes with the 61/11/18 training/validation/test house splits specified by the original dataset [11], [34] are used for the object navigation task. For the RoboTHOR Challenge 2020, there is only the object navigation task. The training and evaluation are split into three phases. In the first phase, the agent is trained on 60 simulated apartments and its performance is validated on 15 other simulated apartments. In the second phase, the agent will be evaluated on four simulated apartments and their real-world counterparts, to test its generalisation to the real world. In the last phase, the agent will be evaluated on 10 real-world apartments.

In this section, we build upon existing visual navigation survey papers [11], [23], [74] to include more recent works.

1) Categories: **Point Navigation** has been one of the foundational and more popular tasks [44] in recent visual navigation literature. In point navigation, an agent is tasked to navigate to any position within a certain fixed distance from a specific point [11]. Generally, the agent is initialized at the origin $(0, 0, 0)$ in an environment, and the fixed goal point is specified by 3D coordinates (x, y, z) relative to the origin/initial location [11]. For the task to be completed successfully, the artificial agent would need to possess a diverse range of skillsets such as visual perception, episodic memory construction, reasoning/planning, and navigation. The agent is usually equipped with a GPS and compass that allows it to access to their location coordinates, and implicitly their orientation relative to the goal position [17], [49]. The target's relative goal coordinates can either be static (i.e. given only once, at the beginning of the episode) or dynamic (i.e. given at every time-step) [17]. More recently, with imperfect localization in indoor environments, Habitat Challenge 2020 has moved on to the more challenging task [47] of RGBD-based online localization without the GPS and compass.

There have been many learning-based approaches to point navigation in recent literature. One of the earlier works [74] uses an end-to-end approach to tackle point navigation in a realistic autonomous navigation setting (i.e. unseen environment with no ground-truth maps and no ground-truth agent's poses) with different sensory inputs. The base navigation algorithm is the Direct Future Prediction (DFP) [94] where relevant inputs such as color image, depth map and actions from the four most recent observations are processed by appropriate neural networks (e.g. convolutional networks for sensory inputs) and concatenated to be passed into a two-stream fully connected action-expectation network. The outputs are the future measurement predictions for all actions and future time steps.

The authors also introduce the Belief DFP (BDFP), which is intended to make the DFP's black-box policy more interpretable by introducing an intermediate map-like representation in future measurement prediction. This is inspired by the attention mechanism in neural networks, and successor representations [95], [96] and features [97] in reinforcement learning. Experiments show that the BDFP outperforms the DFP in most cases, classic navigation approaches generally outperform learning-based ones with RGB-D inputs. [98] provides a more modular approach. For point navigation, SplitNet's architecture consists of one visual encoder and multiple decoders for different auxiliary tasks (e.g. egomotion prediction) and the policy. These decoders aim to learn meaningful representations. With the same PPO algorithm [99] and behavioral cloning training, SplitNet can outperform comparable end-to-end methods in previously unseen environments.

Another work presents a modular architecture for simultaneous mapping and target-driven navigation in indoors environments [48]. In this work, the authors build upon MapNet [71] to include 2.5D memory with semantically-informed features and train a LSTM for the navigation policy. They show that this method outperforms a learned LSTM policy without a map [100] in previously unseen environments.

With the introduction of the *Habitat Challenge* in 2019 and its standardized evaluation, dataset and sensor setups, the more recent approaches have been evaluated with the *Habitat Challenge 2019*. The first work comes from the team behind Habitat, and uses the PPO algorithm, the actor-critic model structure and a CNN for producing embeddings for visual inputs. A follow-up work provides an "existence proof" that near-perfect results can be achieved for the point navigation task for agents with a GPS, a compass and huge learning steps (2.5 billion steps as compared to Habitat's first PPO work with 75 million steps) in unseen environments in simulations [47]. Specifically, the best agent's performance is within 3-5$\%$ of the shortest path oracle. This work uses a modified PPO with Generalized Advantage Estimation [101] algorithm that is suited for distributed reinforcement learning in resource-intensive simulated environments, namely the Decentralized Distributed Proximal Policy Optimization (DD-PPO). At every time-step, the agent receives an egocentric observation (depth or RGB), gets embeddings with a CNN, utilizes its GPS and compass to update the target position to be relative to its current position, then finally outputs the next action and an estimate of the value function. The experiments show that the agents continue to improve for a long time, and the results nearly match that of a shortest-path oracle.

The next work aims to improve on this resource-intensive work by increasing sample and time efficiency with auxiliary tasks [49]. Using the same DD-PPO baseline architecture from the previous work, this work adds three auxiliary tasks: action-conditional contrastive predictive coding (CPC—A) [102], inverse dynamics [68] and temporal distance estimation. The authors experiment with different ways of combining the representations. At 40 million frames, the best performing agent achieves the same performance as the previous work $5.5X$ faster and even has improved performance. The winner of the Habitat Challenge 2019 for both the RGB and the RGB-D tracks [44] provides a hybrid solution that combines both classic and learning-based approaches as end-to-end learning-based approaches are computationally expensive. This work incorporates learning in a modular fashion into a "classic navigation pipeline", thus implicitly incorporating the knowledge of obstacle avoidance and control in low-level navigation. The architecture consists of a learned Neural SLAM module, a global policy, a local policy and an analytical path planner. The Neural SLAM module predicts a map and agent pose estimate using observations and sensors. The global policy always outputs the target coordinates as the long-term goal, which is converted to a short-term goal using the analytic path planner. Finally, a local policy is trained to navigate to this short-term goal. The modular design and use of analytical planning help to reduce the search space during training significantly.

Object Navigation is one of the most straightforward tasks, yet one of the most challenging tasks in embodied AI. Object navigation focuses on the fundamental idea of navigating to an object specified by its label in an unexplored environment [38]. The agent will be initialized at a random position and will be tasked to find an instance of an object category within that environment. Object navigation is generally more complex than point navigation, since it not only requires many of the same skillsets such as visual perception and episodic memory construction, but also semantic understanding. These are what makes the object navigation task much more challenging, but also rewarding to solve.

The task of object navigation can be demonstrated or learnt through adapting, which helps to generalize navigation in an environment without any direct supervision. This work [51] achieves that through a meta-reinforcement learning approach, as the agent learns a self-supervised interaction loss which helps to encourage effective navigation. Unlike the conventional navigation approaches for which the agents freeze the learning model during inference, this work allows the agent learns to adapt itself in a self-supervised manner and adjust or correct its mistake afterwards. This approach prevents an agent from making too many mistakes before realizing and make the necessary correction. Another method is to learn the object relationship between objects before executing the planning of navigation. This work [53] implements an object relation graph (ORG) which is not from external prior knowledge but rather a knowledge graph that is built during the visual exploration phase. The graph consists of object relationships such as category closeness and spatial correlations.

Navigation with Priors focuses on the idea of injecting semantic knowledge or priors in the form of multimodal inputs such as knowledge graph or audio input or to aid in the training of navigation tasks for embodied AI agents in both seen and unseen environments. Past work [57] that use human priors of knowledge integrated into a deep reinforcement learning framework has shown that artificial agent can tap onto human-like semantic/functional priors to aid the agent in learning to navigate and find unseen objects in the unseen environment. Such example taps onto the understanding that the items of interest, such as finding an apple in the kitchen, humans will tend to look at logical locations to begin our search. These knowledge are encoded in a graph network and trained upon in a deep reinforcement learning framework.

There are other examples of using human priors such as human's ability to perceive and capture correspondences between an audio signal modal and the physical location of objects hence to perform navigation to the source of the signal. In this work [103], artificial agents pick multiple sensory observations such as vision and sound signal of the target objects and figure out the shortest trajectory to navigation from its starting location to the source of the sounds. This work achieves it through having a visual perception mapper, sound perception module and dynamic path planners.

Vision-and-Language Navigation (VLN) is a task where agents learn to navigate the environment by following natural language instructions. The challenging aspect of this task is to perceive both the visual scene and language sequentially. VLN remains a challenging task as it requires agents to make predictions of future actions based on past actions and instructions [11]. Furthermore, agents might not be able to align their trajectories seamlessly with natural language instructions. Although vision-and-language navigation and visual question answering (VQA) might seem similar, there are major differences in both tasks. Both tasks can be formulated as visually grounded, sequence-to-sequence transcoding problems. However, VLN sequences are much longer and require vision data to be constantly fed as input and the ability to manipulate camera viewpoints, as compared to VQA where a single input question is fed in and an answer is generated. We are now able to give a natural language instruction to a robot and expect them to perform the task [2], [3], [58]. These are achieved with the advancement of recurrent neural network methods [58] for joint interpretation of both visual and natural language inputs and datasets that are designed for simplifying processes of task-based instruction in navigation and performing of tasks in the 3D environment.

One approach for VLN is the Auxiliary Reasoning Navigation framework [59]. It tackles four auxiliary reasoning tasks: trajectory retelling, progress estimation, angle prediction and cross-modal matching. The agent learns to reason about the previous actions and predicts future information the tasks.

Vision-dialog navigation is the latest extension of VLN as it aims to train an agent to develop the ability to engage in a constant natural language conversation with humans to aid in its navigation. The current work [60] in this area uses a Cross-modal Memory Network (CMN) that remembers and understands useful information related to past navigation actions through separate language memory and visual memory modules, and further uses it to make decisions for navigation.

2) Evaluation Metrics: Visual navigation uses (1) success weighted by (normalized inverse) path length (SPL) and (2) success rate as the main evaluation metrics [11]. Success weighted by path length can be defined as: $\frac{1}{N}\sum_{i=1}^{N} S_i \frac{l_i}{max(p_i,l_i)}$. S_i is a success indicator for episode i, p_i is the agent's path length, l_i is the shortest path length and N is the number of episodes. It is noteworthy that there are some known issues with success weighted by path length [38]. Success rate is the fraction of the episodes in which the agent reaches the goal within the time budget [74]. There are also other less common evaluation metrics [11], [48], [52], [54], [74] in addition to the two mentioned, namely: (3) path length ratio, which is the ratio between the predicted path and the shortest path length and is calculated only for successful episodes; (4) distance to success/navigation error, which measures the distance between the agent's final position and the success threshold boundary around the nearest object or the goal location respectively.

Besides the above four metrics, there are another two metrics used to evaluate VLN agents. They are: (1) oracle success rate, the rate for which the agent stops at the closest point to the goal along its trajectory; (2) trajectory length. In general, for VLN tasks, the best metric is still SPL as it takes into account of the path taken and not just the goal.

For vision-dialog navigation, in addition to success rate and oracle success rate, there are another two metrics used: (1) goal progress, the average agent progress towards the goal location; (2) oracle path success rate, the success rate of agent stopping at the closest point to goal along the shortest path.

3) Datasets: As in visual exploration, Matterport3D and Gibson V1 are the most popular datasets. It is noteworthy that the scenes in Gibson V1 are smaller and usually have shorter episodes (lower GDSP from start position to goal position). The AI2-THOR simulator/dataset is also used.

Unlike the rest of the visual navigation tasks, VLN requires a different kind of dataset. Most of the VLN works use the Room-to-Room (R2R) dataset with the Matterport3D Simulator [104]. It consists of 21,567 navigation instructions with an average length of 29 words. In vision-dialog navigation [59], the Cooperative Vision-and-Dialog Navigation (CVDN) [105] dataset is used. It comprises 2,050 human-to-human dialogs and over 7,000 trajectories within the Matterport3D Simulator.

C. Embodied Question Answering

The task of embodied question answering (QA) in recent embodied AI simulators has been a significant advancement in the field of general-purpose intelligence systems. To perform QA in a state of physical embodiment, an AI agent would need to possess a wide range of AI capabilities such as visual recognition, language understanding, question answering, commonsense reasoning, task planning, and goal-driven navigation. Hence, embodied QA can be considered the most onerous and complicated task in embodied AI research currently.

1) Categories: For **embodied QA** (EQA), a common framework that divides the task into two sub-tasks: a navigation task and a QA task. The navigation module is essential since the agent needs to explore the environment to see the objects before answering questions about them. For example, [61] proposed the Planner-Controller Navigation Module (PACMAN), which comprises a hierarchical structure for the navigation module, with a planner that selects actions (directions) and a controller that decides how far to move following each action. Once the agent decide to stop, the QA module is executed by using the sequence of frames along its path. The navigation module and visual question answering module are first trained individually and then jointly trained by REINFORCE [106]. [62] and [63] further improved the PACMAN model with the Neural Modular Control (NMC) where the higher-level master policy proposes semantic sub-goals to be executed by sub-policies.

Multi-target embodied QA (MT-EQA) [63] is a more complex embodied QA task, which studies questions that have multiple targets in them, e.g. ”Is the apple in the bedroom bigger than the orange in the living room?”, such that the agent has to navigate to the ”bedroom” and the ”living room” to localize the ”apple” and the ”orange” and then perform comparisons to answer the questions.

Interactive Question Answering (IQA) [64] is another work tackling the task of embodied QA in the AI2-THOR environment. IQA is an extension of EQA because it is essential for the agent to interact with the objects to answer certain questions successfully (e.g. the agent needs to open the refrigerator to answer the existence question ”Is there an egg in the fridge?”). [64] proposed using a Hierarchical Interactive Memory Network (HIMN), which is a hierarchy of controllers that help the system operate, learn and reason across multiple time scales, while simultaneously reducing the complexity of each sub-task. An Egocentric Spatial Gated Recurrent Unit (GRU) acts as a memory unit for retaining spatial and semantic information of the environment. The planner module will have control over the other modules such as a navigator which runs an A* search to find the shortest path to the goal, a scanner which performs rotation for detecting new images, a manipulator that is invoked to carry out actions to change the state of the environment and lastly an answerer that will answer the question posted to the AI agent. [65] studied IQA from a multi-agent perspective, where several agents explore an interactive scene jointly to answer a question. [65] proposed multi-layer structural and semantic memories as scene memories to be shared by multiple agents to first reconstruct the 3D scenes and then perform QA.

2) Evaluation Metrics: Embodied QA and IQA involve two sub-tasks: *1) navigation*, and *2) question answering*, and these two sub-tasks are evaluated based on different metrics.

Navigation performance is evaluated by: (1) distance to target at navigation termination, i.e. navigation error (d_T); (2) change in distance to target from initial to final position, i.e. goal progress (d_Δ); (3) smallest distance to target at any point in the episode (d_{min}); (4) percentage of episodes agent terminates navigation for answering before reaching the maximum episode length ($\%stop$); (5) percentage of questions where the agent terminates in the room containing the target object ($\%r_T$); (6) percentage of questions where the agent enters the room containing the target object at least once ($\%r_e$); (7) Intersection over Union for target object (IoU);

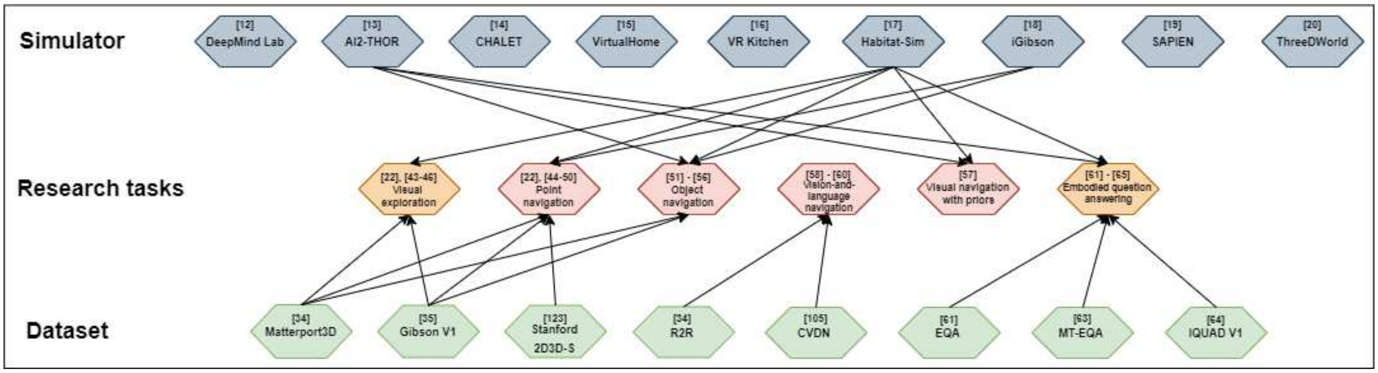

Fig. 6. Connections between Embodied AI simulators to research. (Top) Nine up-to-date embodied AI simulators. (Middle) The various embodied AI research tasks as a result of the nine embodied AI simulators. The red colored research tasks are grouped under the visual navigation category while the rest of the yellow colored tasks are the other research categories. (Bottom) The evaluation dataset used in the evaluation of the research tasks in one of the nine embodied AI simulators..

(8) hit accuracy based on IoU (h_T); (9) episode length, i.e. trajectory length. Metrics (1), (2) and (9) are also used as evaluation metrics for the visual navigation task.

QA performance is evaluated by: (1) mean rank (MR) of the ground-truth answer in predictions; (2) accuracy.

3) Datasets: The EQA [61] dataset is based on House3D, a subset of the popular SUNCG [33] dataset with synthesized rooms and layouts that is similar to the Replica dataset [107]. House3D converts SUNCG's static environment into a virtual environment, where the agent can navigate with physical constraints (e.g. it cannot pass through walls or objects). To test the agent's capabilities in language grounding, commonsense reasoning and navigation, [61] uses a series of functional programs in CLEVR [108] to synthesize questions and answers regarding objects and their properties (e.g. color, existence, location and relative preposition). In total, there are 5,000 questions in 750 environments with reference to 45 unique objects in 7 unique room types.

For MT-EQA [63], the authors introduce the MT-EQA dataset, which contains 6 types of compositional questions which compare object attribute properties (color, size, distance) between multiple targets (objects/rooms).

For IQA [64], the authors annotated a large scale dataset, IQUAD V1, which consist of 75,000 multiple-choice questions. Similar to the EQA dataset, IQUAD V1 has questions regarding object existence, counting and spatial relationships.

IV. Insights and Challenges

A. Insights into Embodied AI

The interconnections in Fig. 6 reflects the suitability of simulators to research tasks. Based on Fig. 6, both Habitat-Sim and iGibson support research tasks in visual exploration and a range of visual navigation tasks, indicating the importance of high fidelity, which comes from *world-based scene* simulators. However, because of their distinct unique features that make them preferable for non-embodied AI standalone tasks such as in deep reinforcement learning, some simulators do not presently connect to any of the embodied research tasks. Nonetheless, they still meet the criteria for being classified as embodied AI simulators.

On the contrary, research tasks such as embodied question answering and visual navigation with priors would require the embodied AI simulators to have *multiple-state* object property, due to the interactive nature of these tasks. Hence, AI2-THOR is undoubtedly the simulator of choice. Lastly, VLN is the only research task that currently does not utilize any of the nine embodied AI simulators but instead uses Matterport3D Simulator [104]. This is because previous works in VLN does not require the feature of *interactivity* in its simulator; hence Matterport3D simulator suffice. However, with the furtherance of VLN tasks, we can expect the need for interactions in VLN tasks, hence the need to use embodied AI simulators. Furthermore, unlike traditional reinforcement learning simulation environments [41], [109] focus on task specific training, while embodied AI simulators provide a training environment for training a wide range of different tasks akin to those undertaken in the physical world.

Furthermore, based on the survey done on the embodied AI research tasks in section III, we propose a pyramid structure in which each embodied AI research task contributes to the next. Visual exploration, for example, aids in the development of visual navigation, and visual navigation contributes to the creation of embodied QA. This build-up approach also correlates with the increasing complexity of the tasks. Based on the foreseeable trends in embodied AI research, we hypothesize that a next advancement in the pyramid of embodied AI research is **Task-based Interactive Question Answering (TIQA)**, which aims to integrate tasks with answering specific questions. For example, such questions can be "*How long would it take for an egg to boil? Is there an apple in the cabinet?*". These are questions that cannot be answered through the conventional approaches [61], [64]. They require the embodied agent to perform specific tasks related to the questions to unlock new insights that are momentous in answering those QA questions. The **TIQA** agents that we hypothesize can perform an array of general household tasks, which allows them to extrapolate useful environmental information that is crucial in helping them to derive the answer to the QA questions. TIQA may hold the key to generalizing task-planning and developing general-purpose AI in simulations which later can be deployed into the real world.

B. Challenges in Embodied AI Simulators

Current embodied AI simulators have reached a level in both its functionality and fidelity, that sets them apart from those conventional simulation used for reinforcement learning. Even with this soaring variance of embodied AI simulators, there are several existing challenges in embodied AI simulators in areas ranging from their **realism**, **scalability** to **interactivity**.

Realism. It focuses on the *fidelity* and *physics* features of the simulators. Simulators with both a high visual fidelity and realistic physics are highly sought after by the robotics communities as they provide the ideal test-bed for various robotic tasks such as navigation and interaction tasks [110], [111]. However, there is a lack of embodied AI simulators that possess both of *world-based scene* and *advanced physics*.

For *fidelity*, simulators that are *world-based scene* will undoubtedly outperform*game-based scene* simulator in simulation to real tasks [27], [112]. Despite this observation, only Habitat-Sim [17], and iGibson [18] are *world-based scene* simulators. This paucity of *world-based scene* simulators is the bottleneck to simulation-to-real tasks for embodied AI agents, which further hinders the transferability of embodied AI research into real-world deployment. For *physics*, the furtherance of physics-based predictive models [113]–[115] have accentuate on the importance of embodied AI simulators with *advanced physics features* as they serve to provide an ideal testbed for training embodied AI agents to perform tasks with sophisticated physical interactions [2], [3], [116]. Despite the need for an advanced physics-based embodied AI simulator, there is currently only one simulator, ThreeDWorld [20] that fits this criterion. Hence, there is a severe lack of embodied AI simulators with advanced physics features such as cloth, fluid and soft-body physics. We believe that advances in 3D reconstruction techniques and physics engines [117]–[119] will improve the realism of embodied AI.

Scalability. Unlike image-based datasets [7], [120] which can be easily obtained from crowd-sourcing or the internet. The methodologies and tools are scarce for collecting large-scale world-based 3D scene datasets and 3D object assets [121]–[123]. These 3D scene datasets are crucial for the construction of a diverse of embodied AI simulators. Current approaches to collect realistic 3D scene datasets requires scanning of the physical room through photogrammetry [124] such as Matterport 3D scanner, Meshroom [125], or even mobile 3D scanning applications. However, they are not commercially viable for collecting large scale 3D objects and scene scans. This is largely due to 3D scanners that are used for photogrammetry are costly and non-accessible. As such, the bottleneck to scalability lies in developing tools for large scale collection of high fidelity 3D object or scene scans. Hopefully, with the further advancement of 3D learning-based approaches [126], [127] that aims to render 3D object meshes from a single or few images or even through scene generation approach [128], we will be able to scale up the collection process of large scale 3D datasets.

Interactivity. The ability to have fine-grained manipulative interactions with functional objects in the embodied AI simulators are crucial in replicating human-level interactions with real-world objects [129]. Most *game-based scene* simulators [13], [16], [19], [20] provides both fine-grained object manipulation capabilities and symbolic interaction capabilities (e.g. <Pulldown Object X on Y> action) or simply a 'point-and-select'. However, due to the nature of *game-based scene* simulators, many research tasks performed in this environment will opt for its symbolic interaction capabilities as compared to fine-grained object manipulation [3], except for a few that utilize both [2], [130].

On the other end, the agents from *world-based scene* simulators [17], [18] possess the ability for gross motor control instead of the symbolic interaction capabilities. However, the object property of the objects within these simulators being largely *interact-able* on the surface which allows for gross motor control but lacks the *multi-state* object classes which is number of state changes that the object have. Hence, there is a need to strike a balance in both the object functionality in its object property and also the complexity of *action* that the embodied AI agent can perform in the environment.

Undoubtedly, mainstream simulators such as AI2-THOR [13], iGibson [18], and Habitat-Sim [17] do provide an excellent environment for advancing the respective embodied AI research. However, they do have their strengths and limitations to be overcome. With developments in computer graphics and computer vision, and the introduction of innovative real-world datasets, real-to-sim domain adaptation is one of the clear routes for improving embodied AI simulators. The concept of real-to-sim revolves around capturing real-world information such as tactile perception [131], human-level motor control [132] and audio inputs [133] in addition to visual sensory inputs and integrating them for the development of more realistic embodied AI simulators that can effectively bridge the physical and virtual worlds.

C. Challenges in Embodied AI Research

Embodied AI research tasks mark an increase in complexity from "internet AI" to autonomous embodied learning agents in 3D simulated environments with multiple sensor modalities and potentially long trajectories [22], [34]. This has led to **memory** and internal representations of the agent becoming extremely important [11], [22], [56]. Long trajectories and multiple input types also signified the importance of robust memory architecture which allows the agent to focus on the important parts of its environment. In recent years, there has been many different types of memory used, such as recurrent neural networks [47], [49], [51], [56], [58], [61]–[63], attention-based memory architectures, [52], [60], [72], anticipated occupancy maps [45], occupancy maps [22] and semantic maps [43], [46], [48], [64], [65], with some papers having overwhelming emphasis on the novelty of their memory architectures [22], [45], [60], [72]. However, while recurrent neural networks are known to be limited in capturing long-term dependencies in embodied AI [56], [72], it is currently still hard to agree which memory type(s) are better [11] due to the lack of work focusing on memory architectures.

Among embodied AI research tasks, there has also been an increase in complexity, as seen in the progression from visual exploration to VLN and embodied QA where new components like language understanding and QA are added respectively. Each new component leads to exponentially harder and longer training of AI agents, especially since current approaches are often fully learning-based. This phenomenon has led to two promising advancements to reduce the search space and sample complexity while improving robustness: **hybrid approaches** combining classic and learning-based algorithms [44], [74] and **prior knowledge incorporation** [23], [57]. Furthermore, **ablation studies are much harder to manage** [31] for more complex tasks as each new component in embodied AI makes it much harder to test for its contribution to the agent's performance, since it is added onto an existing set of components, and embodied AI simulators vary significantly in features and issues. This is compounded by the fact that research tasks have also increased in number rapidly. As a result, while some fundamental tasks like visual exploration have received more attention and thus have more approaches tackling them, the newer and more niche tasks like MT-EQA are much less addressed. New tasks usually introduce new considerations in important aspects like methods, evaluation metrics [22], input types and model components, shown in Table III, thus requiring even more evaluation than simpler tasks like visual exploration.

Lastly, there is a lack of focus on **multi-agent set-ups**, which contribute useful new tasks [65]. This lack of focus can be attributed to the lack of simulators with multi-agent features until recently. Multi-agent systems for collaboration and communication are prevalent in the real world [134], [135] but currently receive relatively little attention [31]. With an increase in simulators with multi-agent features [13], [20], [55] recently, it remains to be seen whether the multi-agent support (e.g. support for multi-agent algorithms) is sufficient.

Conclusion

Recent advances in embodied AI simulators have been a key driver of progress in embodied AI research. Aiming to understand the trends and gaps in embodied AI simulators and research, this paper provides a contemporary and comprehensive overview of embodied AI simulators and research. The paper surveys nine embodied AI simulators and their connections in serving and driving recent innovations in research tasks for embodied AI. By benchmarking nine embodied AI simulators in terms of seven features, we seek to understand their provision of realism, scalability and interactivity, and hence use in embodied AI research. The three main tasks supporting the pyramid of embodied AI research – visual exploration, visual navigation and embodied QA, are examined in terms of their approaches, evaluation metrics, and datasets. This is to review and benchmark the existing approaches in tackling these categories of embodied AI research tasks in the various embodied AI simulators. Furthermore, this paper allows us to unveil insightful relations between the simulators, datasets, and research tasks. With the aid of this paper, AI researchers new to this field would be able to select the most suitable embodied AI simulators for their research tasks and contribute back to advancing the field of embodied AI.

References

[1] L. Smith and M. Gasser, "The development of embodied cognition: Six lessons from babies," *Artificial life*, vol. 11, no. 1-2, pp. 13–29, 2005.

[2] J. Duan, S. Yu, H. L. Tan, and C. Tan, "Actionet: An interactive end-to-end platform for task-based data collection and augmentation in 3d environment," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 1566–1570.

[3] M. Shridhar, J. Thomason, D. Gordon, Y. Bisk, W. Han, R. Mottaghi, L. Zettlemoyer, and D. Fox, "Alfred: A benchmark for interpreting grounded instructions for everyday tasks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 740–10 749.

[4] R. Pfeifer and F. Iida, "Embodied artificial intelligence: Trends and challenges," in *Embodied artificial intelligence*. Springer, 2004, pp. 1–26.

[5] J. Haugeland, "Artificial intelligence: The very idea, cambridge, ma, bradford," 1985.

[6] R. Pfeifer and J. C. Bongard, "How the body shapes the way we think - a new view on intelligence," 2006.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[10] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton *et al.*, "Mastering the game of go without human knowledge," *nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[11] P. Anderson, A. Chang, D. S. Chaplot, A. Dosovitskiy, S. Gupta, V. Koltun, J. Kosecka, J. Malik, R. Mottaghi, M. Savva *et al.*, "On evaluation of embodied navigation agents," *arXiv preprint arXiv:1807.06757*, 2018.

[12] C. Beattie, J. Z. Leibo, D. Teplyashin, T. Ward, M. Wainwright, H. Küttler, A. Lefrancq, S. Green, V. Valdés, A. Sadik *et al.*, "Deepmind lab," *arXiv preprint arXiv:1612.03801*, 2016.

[13] E. Kolve, R. Mottaghi, W. Han, E. VanderBilt, L. Weihs, A. Herrasti, D. Gordon, Y. Zhu, A. Gupta, and A. Farhadi, "Ai2-thor: An interactive 3d environment for visual ai," *arXiv preprint arXiv:1712.05474*, 2017.

[14] C. Yan, D. Misra, A. Bennnett, A. Walsman, Y. Bisk, and Y. Artzi, "Chalet: Cornell house agent learning environment," *arXiv preprint arXiv:1801.07357*, 2018.

[15] X. Puig, K. Ra, M. Boben, J. Li, T. Wang, S. Fidler, and A. Torralba, "Virtualhome: Simulating household activities via programs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8494–8502.

[16] X. Gao, R. Gong, T. Shu, X. Xie, S. Wang, and S.-C. Zhu, "Vrkitchen: an interactive 3d virtual environment for task-oriented learning," *arXiv preprint arXiv:1903.05757*, 2019.

[17] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik *et al.*, "Habitat: A platform for embodied ai research," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9339–9347.

[18] F. Xia, W. B. Shen, C. Li, P. Kasimbeg, M. E. Tchapmi, A. Toshev, R. Martín-Martín, and S. Savarese, "Interactive gibson benchmark: A benchmark for interactive navigation in cluttered environments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 713–720, 2020.

[19] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang *et al.*, "Sapien: A simulated part-based interactive environment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 097–11 107.

[20] C. Gan, J. Schwartz, S. Alter, M. Schrimpf, J. Traer, J. De Freitas, J. Kubilius, A. Bhandwaldar, N. Haber, M. Sano *et al.*, "Threedworld: A platform for interactive multi-modal physical simulation," *arXiv preprint arXiv:2007.04954*, 2020.

[21] M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling, "The arcade learning environment: An evaluation platform for general agents," *Journal of Artificial Intelligence Research*, vol. 47, pp. 253–279, 2013.

[22] S. K. Ramakrishnan, D. Jayaraman, and K. Grauman, "An exploration of embodied visual exploration," *International Journal of Computer Vision*, 2021. [Online]. Available: https://doi.org/10.1007/s11263-021-01437-z

[23] X. Ye and Y. Yang, "From seeing to moving: A survey on learning for visual indoor navigation (vin)," *arXiv preprint arXiv:2002.11310*, 2020.

[24] T. Chen, S. Gupta, and A. Gupta, "Learning exploration policies for navigation," *arXiv preprint arXiv:1903.01959*, 2019.

[25] N. Savinov, A. Dosovitskiy, and V. Koltun, "Semi-parametric topological memory for navigation," in *International Conference on Learning Representations (ICLR)*, 2018.

[26] E. Beeching, J. Dibangoye, O. Simonin, and C. Wolf, "Learning to plan with uncertain topological maps," *arXiv preprint arXiv:2007.05270*, 2020.

[27] A. Kadian, J. Truong, A. Gokaslan, A. Clegg, E. Wijmans, S. Lee, M. Savva, S. Chernova, and D. Batra, "Sim2real predictivity: Does evaluation in simulation predict real-world performance?" *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6670–6677, 2020.

[28] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 1–8.

[29] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 23–30.

[30] CVPR, "Embodied ai workshop," https://embodied-ai.org/, Jan. 2020.

[31] L. Weihs, J. Salvador, K. Kotar, U. Jain, K.-H. Zeng, R. Mottaghi, and A. Kembhavi, "Allenact: A framework for embodied ai research," *arXiv*, 2020.

[32] K. Mo, S. Zhu, A. X. Chang, L. Yi, S. Tripathi, L. J. Guibas, and H. Su, "Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 909–918.

[33] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," *Proceedings of 30th IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[34] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niebner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," in *2017 International Conference on 3D Vision (3DV)*. IEEE Computer Society, 2017, pp. 667–676.

[35] F. Xia, A. R. Zamir, Z. He, A. Sax, J. Malik, and S. Savarese, "Gibson env: Real-world perception for embodied agents," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9068–9079.

[36] Abhishek Kadian*, Joanne Truong*, A. Gokaslan, A. Clegg, E. Wijmans, S. Lee, M. Savva, S. Chernova, and D. Batra, "Are We Making Real Progress in Simulated Environments? Measuring the Sim2Real Gap in Embodied Visual Navigation," in *arXiv:1912.06321*, 2019.

[37] M. Deitke, W. Han, A. Herrasti, A. Kembhavi, E. Kolve, R. Mottaghi, J. Salvador, D. Schwenk, E. VanderBilt, M. Wallingford *et al.*, "Robothor: An open simulation-to-real embodied ai platform," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3164–3174.

[38] D. Batra, A. Gokaslan, A. Kembhavi, O. Maksymets, R. Mottaghi, M. Savva, A. Toshev, and E. Wijmans, "Objectnav revisited: On evaluation of embodied agents navigating to objects," *arXiv preprint arXiv:2006.13171*, 2020.

[39] U. Jain, L. Weihs, E. Kolve, A. Farhadi, S. Lazebnik, A. Kembhavi, and A. Schwing, "A cordial sync: Going beyond marginal policies for multi-agent embodied tasks supplementary material."

[40] U. Jain, L. Weihs, E. Kolve, M. Rastegari, S. Lazebnik, A. Farhadi, A. G. Schwing, and A. Kembhavi, "Two body problem: Collaborative visual task completion," in *CVPR*, 2019, first two authors contributed equally.

[41] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.

[42] U. Jain, L. Weihs, E. Kolve, M. Rastegari, S. Lazebnik, A. Farhadi, A. G. Schwing, and A. Kembhavi, "Two body problem: Collaborative visual task completion," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6689–6699.

[43] D. S. Chaplot, H. Jiang, S. Gupta, and A. Gupta, "Semantic curiosity for active visual learning," in *ECCV*, 2020.

[44] D. S. Chaplot, D. Gandhi, S. Gupta, A. Gupta, and R. Salakhutdinov, "Learning to explore using active neural slam," in *International Conference on Learning Representations (ICLR)*, 2020.

[45] S. K. Ramakrishnan, Z. Al-Halah, and K. Grauman, "Occupancy anticipation for efficient exploration and navigation," *arXiv preprint arXiv:2008.09285*, 2020.

[46] M. Narasimhan, E. Wijmans, X. Chen, T. Darrell, D. Batra, D. Parikh, and A. Singh, "Seeing the un-scene: Learning amodal semantic maps for room navigation," in *European Conference on Computer Vision*. Springer, 2020, pp. 513–529.

[47] E. Wijmans, A. Kadian, A. Morcos, S. Lee, I. Essa, D. Parikh, M. Savva, and D. Batra, "Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames," *arXiv*, pp. arXiv–1911, 2019.

[48] G. Georgakis, Y. Li, and J. Kosecka, "Simultaneous mapping and target driven navigation," *arXiv preprint arXiv:1911.07980*, 2019.

[49] J. Ye, D. Batra, E. Wijmans, and A. Das, "Auxiliary tasks speed up learning pointgoal navigation," *arXiv preprint arXiv:2007.04561*, 2020.

[50] C. Pérez-D'Arpino, C. Liu, P. Goebel, R. Martín-Martín, and S. Savarese, "Robot navigation in constrained pedestrian environments using reinforcement learning," *arXiv preprint arXiv:2010.08600*, 2020.

[51] M. Wortsman, K. Ehsani, M. Rastegari, A. Farhadi, and R. Mottaghi, "Learning to learn how to learn: Self-adaptive visual navigation using meta-learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6750–6759.

[52] T. Campari, P. Eccher, L. Serafini, and L. Ballan, "Exploiting scene-specific features for object goal navigation," in *ECCV Workshops*, 2020.

[53] H. Du, X. Yu, and L. Zheng, "Learning object relation graph and tentative policy for visual navigation," in *European Conference on Computer Vision*. Springer, 2020, pp. 19–34.

[54] D. S. Chaplot, D. Gandhi, A. Gupta, and R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," *arXiv preprint arXiv:2007.00643*, 2020.

[55] B. Shen, F. Xia, C. Li, R. Martın-Martın, L. Fan, G. Wang, S. Buch, C. D'Arpino, S. Srivastava, L. P. Tchapmi, K. Vainio, L. Fei-Fei, and S. Savarese, "igibson, a simulation environment for interactive tasks in large realistic scenes," *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[56] A. Wahid, A. Stone, K. Chen, B. Ichter, and A. Toshev, "Learning object-conditioned exploration using distributed soft actor critic," *arXiv preprint arXiv:2007.14545*, 2020.

[57] W. Yang, X. Wang, A. Farhadi, A. Gupta, and R. Mottaghi, "Visual semantic navigation using scene priors," *arXiv preprint arXiv:1810.06543*, 2018.

[58] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, and A. 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*, 2018, pp. 3674–3683.

[59] F. Zhu, Y. Zhu, X. Chang, and X. Liang, "Vision-language navigation with self-supervised auxiliary reasoning tasks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 012–10 022.

[60] Y. Zhu, F. Zhu, Z. Zhan, B. Lin, J. Jiao, X. Chang, and X. Liang, "Vision-dialog navigation by exploring cross-modal memory," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 730–10 739.

[61] A. Das, S. Datta, G. Gkioxari, S. Lee, D. Parikh, and D. Batra, "Embodied question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 2054–2063.

[62] A. Das, G. Gkioxari, S. Lee, D. Parikh, and D. Batra, "Neural modular control for embodied question answering," in *Proceedings of the Conference on Robot Learning (CoRL)*, 2018.

[63] L. Yu, X. Chen, G. Gkioxari, M. Bansal, T. L. Berg, and D. Batra, "Multi-target embodied question answering," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 2019, pp. 6309–6318.

[64] D. Gordon, A. Kembhavi, M. Rastegari, J. Redmon, D. Fox, and A. Farhadi, "Iqa: Visual question answering in interactive environments," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4089–4098.

[65] S. Tan, W. Xiang, H. Liu, D. Guo, and F. Sun, "Multi-agent embodied question answering in interactive environments," in *ECCV 2020 - 16th European Conference, Glasgow,UK, August 23-28, 2020, Proceedings*, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds., 2020, pp. 663–678.

[66] P. D. Nguyen, Y. K. Georgie, E. Kayhan, M. Eppe, V. V. Hafner, and S. Wermter, "Sensorimotor representation learning for an "active self" in robots: a model survey," *KI-Künstliche Intelligenz*, vol. 35, no. 1, pp. 9–35, 2021.

[67] C. Chen, U. Jain, C. Schissler, S. V. A. Gari, Z. Al-Halah, V. K. Ithapu, P. Robinson, and K. Grauman, "Soundspaces: Audio-visual navigation in 3d environments," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*. Springer, 2020, pp. 17–36.

[68] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven exploration by self-supervised prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 16–17.

[69] S. Gupta, D. Fouhey, S. Levine, and J. Malik, "Unifying map and landmark based representations for visual navigation," *arXiv preprint arXiv:1712.08125*, 2017.

[70] S. Gupta, J. Davidson, S. Levine, R. Sukthankar, and J. Malik, "Cognitive mapping and planning for visual navigation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2616–2625.

[71] J. F. Henriques and A. Vedaldi, "Mapnet: An allocentric spatial memory for mapping environments," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8476–8484.

[72] K. Fang, A. Toshev, L. Fei-Fei, and S. Savarese, "Scene memory transformer for embodied agents in long-horizon tasks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 538–547.

[73] L. Mezghani, S. Sukhbaatar, A. Szlam, A. Joulin, and P. Bojanowski, "Learning to visually navigate in photorealistic environments without any supervision," *arXiv preprint arXiv:2004.04954*, 2020.

[74] D. Mishkin, A. Dosovitskiy, and V. Koltun, "Benchmarking classic and learned navigation in complex 3d environments," *arXiv preprint arXiv:1901.10915*, 2019.

[75] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[76] S. K. Ramakrishnan, D. Jayaraman, and K. Grauman, "Emergence of exploratory look-around behaviors through active observation completion," *Science Robotics*, vol. 4, no. 30, 2019.

[77] W. S. Lovejoy, "A survey of algorithmic methods for partially observed markov decision processes," *Annals of Operations Research*, vol. 28, no. 1, pp. 47–65, 1991.

[78] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97.'Towards New Computational Principles for Robotics and Automation'*. IEEE, 1997, pp. 146–151.

[79] Y. Burda, H. Edwards, D. Pathak, A. Storkey, T. Darrell, and A. A. Efros, "Large-scale study of curiosity-driven learning," *arXiv preprint arXiv:1808.04355*, 2018.

[80] R. Houthooft, X. Chen, Y. Duan, J. Schulman, F. De Turck, and P. Abbeel, "Vime: Variational information maximizing exploration," in *Advances in Neural Information Processing Systems*, 2016, pp. 1109–1117.

[81] D. Pathak, D. Gandhi, and A. Gupta, "Self-supervised exploration via disagreement," in *ICML*, 2019.

[82] Y. Burda, H. Edwards, A. Storkey, and O. Klimov, "Exploration by random network distillation," *arXiv preprint arXiv:1810.12894*, 2018.

[83] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[84] S. K. Ramakrishnan and K. Grauman, "Sidekick policy learning for active visual exploration," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 413–430.

[85] S. Song, A. Zeng, A. X. Chang, M. Savva, S. Savarese, and T. Funkhouser, "Im2pano3d: Extrapolating 360 structure and semantics beyond the field of view," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3847–3856.

[86] N. Savinov, A. Raichuk, R. Marinier, D. Vincent, M. Pollefeys, T. Lillicrap, and S. Gelly, "Episodic curiosity through reachability," *arXiv preprint arXiv:1810.02274*, 2018.

[87] N. Haber, D. Mrowca, S. Wang, L. F. Fei-Fei, and D. L. Yamins, "Learning to play with intrinsically-motivated, self-aware agents," in *Advances in Neural Information Processing Systems*, 2018, pp. 8388–8399.

[88] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 3357–3364.

[89] D. S. Chaplot, R. Salakhutdinov, A. Gupta, and S. Gupta, "Neural topological slam for visual navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 875–12 884.

[90] F. Bonin-Font, A. Ortiz, and G. Oliver, "Visual navigation for mobile robots: A survey," *Journal of intelligent and robotic systems*, vol. 53, no. 3, p. 263, 2008.

[91] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha, "Visual simultaneous localization and mapping: a survey," *Artificial intelligence review*, vol. 43, no. 1, pp. 55–81, 2015.

[92] 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*, vol. 12, no. 4, pp. 566–580, 1996.

[93] S. M. LaValle and J. J. Kuffner, "Rapidly-exploring random trees: Progress and prospects," *Algorithmic and computational robotics: new directions*, no. 5, pp. 293–308, 2001.

[94] A. Dosovitskiy and V. Koltun, "Learning to act by predicting the future," *arXiv preprint arXiv:1611.01779*, 2016.

[95] P. Dayan, "Improving generalization for temporal difference learning: The successor representation," *Neural Computation*, vol. 5, no. 4, pp. 613–624, 1993.

[96] Y. Zhu, D. Gordon, E. Kolve, D. Fox, L. Fei-Fei, A. Gupta, R. Mottaghi, and A. Farhadi, "Visual semantic planning using deep successor representations," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 483–492.

[97] A. Barreto, W. Dabney, R. Munos, J. J. Hunt, T. Schaul, H. P. van Hasselt, and D. Silver, "Successor features for transfer in reinforcement learning," *Advances in neural information processing systems*, vol. 30, pp. 4055–4065, 2017.

[98] D. Gordon, A. Kadian, D. Parikh, J. Hoffman, and D. Batra, "Splitnet: Sim2sim and task2task transfer for embodied visual navigation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1022–1031.

[99] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms. arxiv 2017," *arXiv preprint arXiv:1707.06347*.

[100] A. Mousavian, A. Toshev, M. Fišer, J. Košecká, A. Wahid, and J. Davidson, "Visual representations for semantic target driven navigation," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8846–8852.

[101] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," *arXiv preprint arXiv:1506.02438*, 2015.

[102] Z. D. Guo, M. G. Azar, B. Piot, B. A. Pires, and R. Munos, "Neural predictive belief representations," *arXiv preprint arXiv:1811.06407*, 2018.

[103] C. Gan, Y. Zhang, J. Wu, B. Gong, and J. B. Tenenbaum, "Look, listen, and act: Towards audio-visual embodied navigation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9701–9707.

[104] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, and A. 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 (CVPR)*, 2018.

[105] J. Thomason, M. Murray, M. Cakmak, and L. Zettlemoyer, "Vision-and-dialog navigation," in *Conference on Robot Learning (CoRL)*, 2019.

[106] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, vol. 8, no. 3-4, pp. 229–256, 1992.

[107] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. J. Engel, R. Mur-Artal, C. Ren, S. Verma *et al.*, "The replica dataset: A digital replica of indoor spaces," *arXiv preprint arXiv:1906.05797*, 2019.

[108] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick, "Clevr: A diagnostic dataset for compositional language and elementary visual reasoning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2901–2910.

[109] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 5026–5033.

[110] C. Gao, J. Chen, S. Liu, L. Wang, Q. Zhang, and Q. Wu, "Room-and-object aware knowledge reasoning for remote embodied referring

expression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3064–3073.

[111] J. Sun, D.-A. Huang, B. Lu, Y.-H. Liu, B. Zhou, and A. Garg, "Plate: Visually-grounded planning with transformers in procedural tasks," 2021.

[112] F. Sadeghi and S. Levine, "Cad2rl: Real single-image flight without a single real image," *Robotics: Science and Systems XIII, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA, July 12-16, 2017*, 2017.

[113] D. M. Bear, E. Wang, D. Mrowca, F. J. Binder, H.-Y. F. Tung, R. Pramod, C. Holdaway, S. Tao, K. Smith, L. Fei-Fei *et al.*, "Physion: Evaluating physical prediction from vision in humans and machines," *arXiv preprint arXiv:2106.08261*, 2021.

[114] J. Duan, S. Y. B. Jian, and C. Tan, "Space: A simulator for physical interactions and causal learning in 3d environments," *arXiv preprint arXiv:2108.06180*, 2021.

[115] J. Duan, S. Yu, S. Poria, B. Wen, and C. Tan, "Pip: Physical interaction prediction via mental imagery with span selection," *arXiv preprint arXiv:2109.04683*, 2021.

[116] T. Nagarajan and K. Grauman, "Learning affordance landscapes for interaction exploration in 3d environments," *Advances in Neural Information processing Systems 33*, 2020.

[117] M. Wang, Y. Deng, X. Kong, A. H. Prasad, S. Xiong, and B. Zhu, "Thin-film smoothed particle hydrodynamics fluid," *arXiv preprint arXiv:2105.07656*, 2021.

[118] A. Kuznetsov, K. Mullia, Z. Xu, M. Hašan, and R. Ramamoorthi, "Neumip: multi-resolution neural materials," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 1–13, 2021.

[119] S. R. Richter, H. A. AlHaija, and V. Koltun, "Enhancing photorealism enhancement," *arXiv preprint arXiv:2105.04619*, 2021.

[120] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[121] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. J. Engel, R. Mur-Artal, C. Ren, S. Verma *et al.*, "The replica dataset: A digital replica of indoor spaces," *arXiv preprint arXiv:1906.05797*, 2019.

[122] S. K. Ramakrishnan, A. Gokaslan, E. Wijmans, O. Maksymets, A. Clegg, J. Turner, E. Undersander, W. Galuba, A. Westbury, A. X. Chang, M. Savva, Y. Zhao, and D. Batra, "Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai," 2021.

[123] I. Armeni, S. Sax, A. R. Zamir, and S. Savarese, "Joint 2d-3d-semantic data for indoor scene understanding," *arXiv preprint arXiv:1702.01105*, 2017.

[124] E. M. Mikhail, J. S. Bethel, and J. C. McGlone, "Introduction to modern photogrammetry," *New York*, vol. 19, 2001.

[125] Alicevision, *Blender - a 3D modelling and rendering package*, Alicevision, 2018. [Online]. Available: https://github.com/alicevision/meshroom

[126] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "pixelnerf: Neural radiance fields from one or few images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4578–4587.

[127] R. Martin-Brualla, N. Radwan, M. S. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth, "Nerf in the wild: Neural radiance fields for unconstrained photo collections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7210–7219.

[128] Z. Hao, A. Mallya, S. Belongie, and M.-Y. Liu, "Gancraft: Unsupervised 3d neural rendering of minecraft worlds," *arXiv preprint arXiv:2104.07659*, 2021.

[129] Y. Zhu, T. Gao, L. Fan, S. Huang, M. Edmonds, H. Liu, F. Gao, C. Zhang, S. Qi, Y. N. Wu *et al.*, "Dark, beyond deep: A paradigm shift to cognitive ai with humanlike common sense," *Engineering*, vol. 6, no. 3, pp. 310–345, 2020.

[130] M. Lohmann, J. Salvador, A. Kembhavi, and R. Mottaghi, "Learning about objects by learning to interact with them," *Advances in Neural Information processing Systems 33*, 2020.

[131] R. Bhirangi, T. Hellebrekers, C. Majidi, and A. Gupta, "Reskin: versatile, replaceable, lasting tactile skins," in *5th Annual Conference on Robot Learning*, 2021.

[132] B. Smith, C. Wu, H. Wen, P. Peluse, Y. Sheikh, J. K. Hodgins, and T. Shiratori, "Constraining dense hand surface tracking with elasticity," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 6, pp. 1–14, 2020.

[133] C. Chen, U. Jain, C. Schissler, S. V. A. Gari, Z. Al-Halah, V. K. Ithapu, P. Robinson, and K. Grauman, "Audio-visual embodied navigation," *environment*, vol. 97, p. 103, 2019.

[134] L. Panait and S. Luke, "Cooperative multi-agent learning: The state of the art," *Autonomous agents and multi-agent systems*, vol. 11, no. 3, pp. 387–434, 2005.

[135] Y.-C. Liu, J. Tian, C.-Y. Ma, N. Glaser, C.-W. Kuo, and Z. Kira, "Who2com: Collaborative perception via learnable handshake communication," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6876–6883.

V. Biography Section

Jiafei Duan received the B.Eng. (Highest Distinction) degree from the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore, in 2021. He is currently working as a research engineer at the Institute of Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore. His current research interest is embodied AI and computational cognitive science.

Samson Yu received a B.Eng. degree in Information Systems Technology and Design from the Singapore University of Technology and Design in 2020. He is currently working as a research engineer at the Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore, on fundamental AI research and embodied AI.

Hui Li Tan received the B.Sc. degree in applied mathematics from the National University of Singapore (NUS), Singapore, in 2007. She received the Ph.D. degree in electrical and computer engineering, from NUS in 2017. Since 2007, she has been with the Institute for Infocomm Research, Singapore. Her current research interests include computer vision, multimodal deep learning, incremental and federated learning.

Hongyuan Zhu received his Ph.D. degree in computer engineering from Nanyang Technologi- cal University, Singapore, in 2014. He is cur- rently a Research Scientist with the Institute for Infocomm Research, A*STAR, Singapore. His research interests include multimedia content analysis and segmentation.

Cheston Tan received B.Sc. (Highest Honours) degree from the Department of Electrical Engineering and Computer Science, University of California, Berkeley, as well as the Ph.D. degree from the Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology. He is currently a senior scientist at the Institute for Infocomm Research (I2R), Agency for Science, Technology and Research (A*STAR), Singapore.