EMBODIED WEB AGENTS: Bridging Physical-Digital Realms for Integrated Agent Intelligence

Yining Hong* **Rui Sun*** **Bingxuan Li†** **Xingcheng Yao†** **Maxine Wu†** **Alexander Chien†**
Da Yin **Ying Nian Wu** **Zhecan James Wang** **Kai-Wei Chang**

University of California, Los Angeles

Abstract

AI agents today are mostly siloed — they either retrieve and reason over vast amount of digital information and knowledge obtained online; or interact with the physical world through embodied perception, planning and action — but rarely both. This separation limits their ability to solve tasks that require integrated physical and digital intelligence, such as cooking from online recipes, navigating with dynamic map data, or interpreting real-world landmarks using web knowledge. We introduce EMBODIED WEB AGENTS, a novel paradigm for AI agents that fluidly bridge embodiment and web-scale reasoning. To operationalize this concept, we first develop the EMBODIED WEB AGENTS task environments, a unified simulation platform that tightly integrates realistic 3D indoor and outdoor environments with functional web interfaces. Building upon this platform, we construct and release the EMBODIED WEB AGENTS Benchmark, which encompasses a diverse suite of tasks including cooking, navigation, shopping, tourism, and geolocation — all requiring coordinated reasoning across physical and digital realms for systematic assessment of cross-domain intelligence. Experimental results reveal significant performance gaps between state-of-the-art AI systems and human capabilities, establishing both challenges and opportunities at the intersection of embodied cognition and web-scale knowledge access. All datasets, codes and websites are publicly available at our project page https://embodied-web-agent.github.io/.

1 Introduction

Recently, we have seen the proliferation of web agents capable of retrieving information online [Shi et al., 2017, Yao et al., 2022, Deng et al., 2023, Zhou et al., 2023, Koh et al., 2024] — yet they remain confined to screens disembodied from the real world. Meanwhile, their physical counterparts — robots and embodied systems — navigate the world but with limited access to the Internet. What if the boundary between the digital and physical realms were shattered? What if web agents stepped out of the browser, with keys to perceive and act in the real 3D physical world, while physical robots autonomously tapped into the encyclopedic knowledge of the web? As illustrated in Figure 1, such agents would not only assess the ingredients in your kitchen, search for matching recipes online, shop for missing items, and cook your favorite dish for you; but also traverse historical landmarks, interpret architectural styles using both their own perception and Wikipedia, leave personalized reviews, and perhaps even return with a souvenir in hand. We, as humans, don't compartmentalize our intelligence into "physical-only" and "digital-only" modules — we fluidly move between realms. What if contemporary AI agents could likewise achieve the best of both worlds?

Building such agents *goes far beyond a mere combination of isolated web and embodied systems*; it presents a set of deeply intertwined challenges. The first is *the perceptual grounding problem*: how can an agent link abstract digital instructions (e.g., "cook potato and egg until golden brown" as in Figure 1 (b)) with the high-dimensional data streams of the physical world (e.g., visually recognizing

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

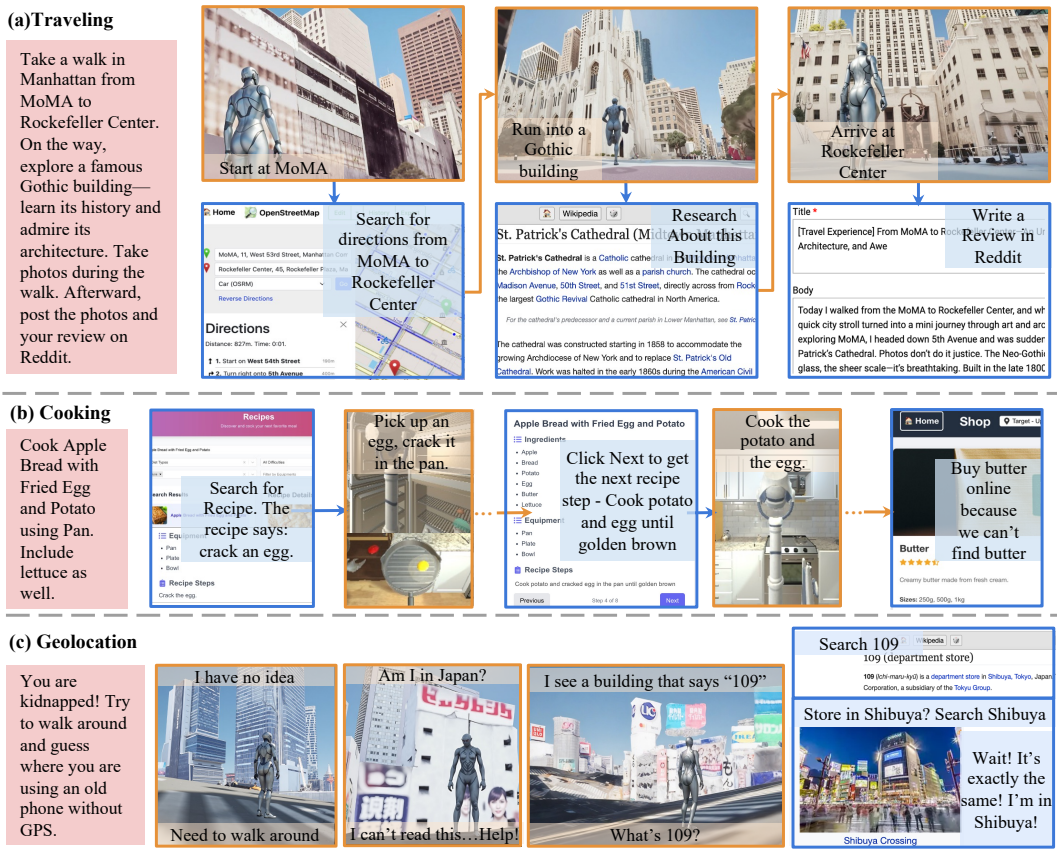

Figure 1: **Illustrative examples of our EMBODIED WEB AGENTS conceptual paradigm, tasks and environments.** Blue boxes and arrows indicate web interaction / switching to the web respectively. Orange boxes and arrows indicates acting in / switching to the embodied environment. We omit most intermediate actions due to the large number of interaction steps.

the transition of potatoes and eggs to a golden state through a series of embodied observations)? Addressing this requires embodied perception, where agents actively interpret their surroundings through movement, interaction, and multimodal sensing — continually acquiring feedback from their environment and aligning these observations with digital instructions. The second challenge is *cross-domain planning*: how should an agent decide when to shift between physical actions and digital information retrieval, particularly when information from one domain contradicts or supplements the other? For instance, the online map may suggest a path to visit Rockefeller Center, but real-world observation may reveal that the center is closed due to a protest, demanding a dynamic reevaluation of the agent's plan. To navigate seamlessly between domains, agents must maintain a coherent and persistent representation that bridges physical and digital contexts — recalling physical experiences when operating online, and retrieving digital knowledge when acting in the world. Despite all these challenges, there remains a surprising lack of research targeting this level of integrated intelligence — both in terms of conceptual frameworks and benchmark development. As a result, progress in each domain often unfolds in isolation, with limited cross-pollination between the two paradigms.

To this end, we introduce EMBODIED WEB AGENTS as a new conceptual paradigm of AI systems that unify physical embodiment with web-scale knowledge access — capable of perceiving and acting in the real world while reasoning over dynamic, unstructured information from the web. To operationalize this concept, we first develop the EMBODIED WEB AGENTS task environments, a unified simulation platform that integrates realistic 3D environments with interactive web interfaces. This platform combines (1) indoor settings from AI2-THOR, (2) outdoor navigation in Google Earth, and (3) web interfaces including Wikipedia, online stores, recipe websites, map services *etc.*, enabling agents to interact seamlessly with both physical and digital spaces. Building upon this environment, we construct the EMBODIED WEB AGENTS Benchmark, which encompasses approximately 1.5k

tasks across multiple domains, including: (1) cooking tasks where agents match physical ingredients with online recipes; (2) navigation combining online maps with physical wayfinding; (3) shopping requiring coordination between in-store actions and online options; (4) tourism connecting physical landmarks with web information; and (5) geolocation determining position through embodied exploration and online research. Together, these tasks systematically test an agent's ability to bridge embodied perception, action, and web-based reasoning across varied contexts.

We conduct comprehensive experiments on our proposed EMBODIED WEB AGENTS benchmark using several state-of-the-art LLM agent baselines, including GPT, Gemini, Qwen, and Intern models. Experimental results show that current LLM agents are far from satisfactory compared to human performances. A detailed breakdown and analysis of error types and their percentage contributions to task failures also reveal that current models predominantly struggle with cross-domain integration, not isolated capabilities. For instance, these models encounter problems such as being trapped in a single environment and unable to switch to the other domain, or the misalignment of web instructions and embodied actions. This further strengthens our position that embodied web agency presents unique challenges that cannot be studied through isolated physical or digital agents alone, as the key difficulties emerge precisely at the intersection where these domains are intertwined.

The key contributions of this paper can be summarized as follows.

- We introduce EMBODIED WEB AGENTS as a new conceptual paradigm for AI systems that integrate embodiment with web-scale information access — formalizing a class of agents capable of acting in the physical world while reasoning over unstructured digital content.
- We develop the EMBODIED WEB AGENTS task environments, a unified simulation platform that tightly integrates realistic 3D environments with interactive web interfaces, enabling agents to perform cross-domain tasks involving perception, action, and retrieval.
- We construct and release the EMBODIED WEB AGENTS Benchmark, which encompasses a diverse suite of tasks across multiple domains including navigation, shopping, traveling, cooking and geolocation.
- We conduct in-depth empirical analysis of state-of-the-art LLM agents on our benchmark, revealing that our benchmark poses rigorous challenges for current LLM agents, and opens up a challenging new direction and testbed for future agents with integrated intelligence.

2 Related Works

Web Agent Benchmarks Web agents are designed to navigate and interact with web environments to complete tasks following user instruction. Initial web agent evaluation benchmarks such as MiniWoB [Shi et al., 2017] and MiniWoB++ [Liu et al., 2018] introduce a suite of diverse web navigation tasks on synthetic webpages. More recent benchmarks emphasize greater realism and task diversity. WebShop [Yao et al., 2022] simulates an e-commerce platform with numbers of products to evaluate agents' ability to search and make purchases, while Mind2Web [Deng et al., 2023] provides a diverse collection of open-ended tasks across hundreds of real websites to assess general web navigation and interaction capabilities. Similarly, benchmarks like WebArena [Zhou et al., 2023], WebVoyager [He et al., 2024], WebLINX [Lù et al., 2024], and VisualWebArena [Koh et al., 2024] feature fully functional websites spanning multiple domains, enabling the evaluation of agents on long-horizon tasks in realistic, diverse environments. OVEN [Hu et al., 2023] challenges models to link images to specific Wikipedia entities given text queries. Beyond pursuing more realistic test environments, WorkArena [Drouin et al., 2024] requires agents to interact with enterprise software and perform tasks demanding higher expertise and comprehension. In this work, we explore a distinct yet important scenario where web browsing is integrated into the physical embodied world.

Embodied Environments and Benchmarks Recent developments in environments and benchmarks have accelerated the research on embodied AI. Simulation platforms, such as AI2-THOR [Kolve et al., 2017], Habitat [Manolis Savva* et al., 2019] and iGibson [Shen et al., 2021, Li et al., 2022], enable agents to perform diverse interactive tasks in realistic indoor environments. Benchmarks like ALFRED [Shridhar et al., 2020] and BEHAVIOR [Srivastava et al., 2021] provide a diverse suite of indoor tasks for embodied agents, requiring instruction understanding, long-horizon planning and manipulation in a closed environment. Additionally, Embodied Agent Inferface [Li et al., 2024] formalizes decision processes for LLM-based embodied agents and introduces fine-grained evaluation metrics for indoor embodied tasks. Efforts have also been made to extend the applicability

Figure 2: **An Exemplar Pipeline of completing a task in our EMBODIED WEB AGENTS dataset.** Blue boxes indicate web interaction. Orange boxes indicate embodied interaction. Boxes with gradient colors indicate switching from one environment to the other.

of embodied agents to outdoor environments. A series of outdoor navigation benchmarks, such as StreetLearn [Mirowski et al., 2018], TouchDown [Chen et al., 2019, Mehta et al., 2020], RUN [Paz-Argaman and Tsarfaty, 2019], have been introduced to evaluate the ability of embodied agents on vision-language navigation and spatial description resolution in urban street environments. Du and Varshney [2016], Du et al. [2019] create immersive systems that integrate geo-tagged social media with 3D street-level environments to enhance virtual and augmented reality experiences for storytelling, tourism, and cultural exploration. Yang et al. [2024] V-IRL is a platform for training and testing AI agents in realistic virtual environments to develop real-world skills. More outdoor related tasks such as geolocation prediction [Haas et al., 2023] and map understanding [Xing et al., 2025] has also been proposed recently. In this work, we design a new benchmark encompassing a diverse set of embodied tasks within both indoor and outdoor environments. Different from previous works, our benchmark focuses on embodied tasks that require web access and interaction to be completed, a realistic scenario that is challenging and neglected in existing benchmarks.

Cross-Modal Agent Systems Cross-modal agent systems integrating vision, language and other modalities have been explored in both web and embodied environments. In web-based settings, He et al. [2024] builds a web agent powered by a large multimodal model that interacts with real-world websites following user instructions. Lin et al. [2024] develops ShowUI, an efficient vision-laguage-action model for GUI agent. For embodied tasks, multimodal foundation models such as Gato [Reed et al., 2022], PaLM-E [Driess et al., 2023] and 3D-LLM [Hong et al., 2023] have been developed to provide generalist policies in real world. In this work, we explore a new dimension for modal fusion in embodied agents, by integrating both embodied and web actions into one unified framework, to enable agents to perform more complex and diverse tasks with real-world applications.

3 The EMBODIED WEB AGENT Task Environments

Inspired by Zhou et al. [2023], our environments are formalized as $E = \langle S, A, O, T \rangle$, where S is the combined physical-digital state space, A is the action space spanning both domains, and O is the observation space comprising embodied input o_t^e and web perception o_t^w. The deterministic transition function $T : S \times A \to S$ governs state evolution as agents select actions based on task specification, observations, and history. Task completion is measured by reward function $r(a_1^T, s_1^T)$ evaluating whether actions successfully fulfill intents like cooking dishes or reaching destinations.

Our task environments can be categorized into three parts: outdoor environment (3.1), indoor environment (3.2) and web environment (3.3). We show an example of interacting with and switching among the environments in Figure 2, as well as the action spaces of all environments in Table 1.

Action	Explanation
INDOOR ENVIRONMENT ACTIONS	
Agent Movement	
`Teleport [obj]`	Teleport agent to a specific object
`MoveAhead/Back/Left/Right`	Move agent in a cardinal direction
Object Interaction	
`PickupObject / PutObject [obj]`	Pick up or put held object
Object State Changes	
`OpenObject / CloseObject [obj]`	Open or close an object
`SliceObject [obj]`	Slice an object
`CookObject [obj]`	Cook an object
Environment Switching	
`switch_environment [msg]`	Switch between web/embodied
OUTDOOR ENVIRONMENT ACTIONS	
`Forward / Left / Right`	Move agent in outdoor environment
WEB ENVIRONMENT ACTIONS	
Page Operation Actions	
`click [id]`	Click on an element with specific id
`type [id] [content] [pr]`	Type content into field
`scroll [direction]`	Scroll page up or down
`hover [id] / press [key_comb]`	Hover or simulate key press
Tab Management & URL Navigation Actions	
`new_tab / close_tab / tab_focus`	Open, close or focus on a tab
`goto [url] / go_back / forward`	Navigate to URL or go back/forward

Table 1: Action Spaces for All Environments

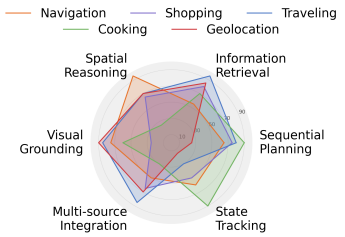

Figure 3: Importance of Different Capabilities Across Tasks

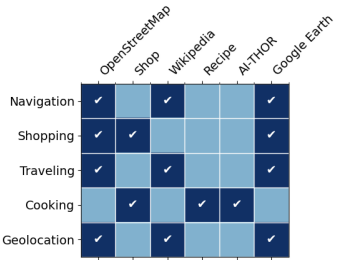

Figure 4: Environments for Tasks

3.1 Outdoor Environment

The outdoor environment is constructed by leveraging the Google Street View and Google Earth API, which provides real-world, street-level observations captured by Google's panoramic cameras. To build the outdoor environment, we select four cities (i.e., New York, Boston, Philadelphia, and Pittsburgh) with visually and structurally complex street layouts. Unlike synthetic or simulation-based environments, the visual data provided by Google is inherently more natural, noisy, and diverse, offering a more challenging and representative benchmark. Through API calls, we retrieve observations associated with specific geographic coordinates. These include panoramic images or standard-perspective images in cardinal directions. Alongside visual data, we also obtain: the GPS coordinates of each point, the heading / directional metadata between connected points, and the connectivity (adjacency) information across locations. With these elements, we construct a navigation graph that underlies the outdoor environment. Formally, this environment can be described as an undirected graph $G = (V, E)$, where each node $v \in V$ represents a specific GPS coordinate, each edge $e \in E$ encodes a connection between two coordinates, including heading and distance, and each node is associated with four directional visual observations (north, east, south, west), represented as standard field-of-view images. Agents interact with the outdoor environment by observing these visual inputs, accessing the neighboring node set, and using heading information to reason about spatial transitions. Given navigation instructions (e.g., derived from web-based directions), the agent must determine which neighbor to move to at each step in order to reach a specified goal location, completing the navigation task through step-by-step decision making. This design closely mirrors real-world settings and introduces challenges that go beyond those posed by synthetic simulators. Compared to environments with simplified or rendered visuals, our outdoor environment demands stronger generalization and robustness from embodied agents, making it a more practical and realistic testbed for evaluating agent systems in open-world scenarios.

3.2 Indoor Environment

The indoor task environment utilizes AI2-THOR [Kolve et al., 2017], a photorealistic 3D indoor simulation platform. The environment provides highly accurate and interactive kitchen scenes containing fresh ingredients, cooking equipment, storage containers, and kitchen appliances. Agents can observe ingredient states, manipulate objects, and monitor cooking progress through visual perception. Objects are tracked with properties and states, including boolean flags (e.g., isSliced, isCooked), location information (e.g., parentReceptacles), and more, all of which dynamically update as agents execute physical actions like chopping or mixing, instructed by online recipes. A specialized state evaluator compares the current kitchen state against ideal target states, measuring task completion by checking whether objects have achieved desired states and spatial arrangements.

3.3 Web Environment

The web environment consists of five functional websites, each supporting different aspects of agent interaction across both indoor and outdoor scenarios. The websites are implemented with a React.js frontend structured using modular components and state management, and a FastAPI backend that exposes asynchronous RESTful APIs for data serving and user interaction. The homepage serves as the central navigation hub, linking to all other task-specific websites and maintaining contextual continuity across interactions. The recipe website we built allows users to browse, search, and filter cooking recipes based on ingredients, dietary preferences, or cuisine types. The shopping website built from scratch enables management of a shopping cart, ingredient lookup, and simulated checkout processes. It facilitates task flows involving item selection, inventory reasoning, and purchasing. We also adapt several websites from the WebArena benchmark [Zhou et al., 2023]. The OpenStreetMap site offers an interactive map for location search, address lookup, and exploration of geographic entities. The Wikipedia site presents richly interlinked encyclopedic content for information-seeking, entity linking, and multi-hop reasoning across documents. These websites are modified slightly to ensure smooth integration with the homepage. All websites are public and can be reached. More details can be found in our project page https://embodied-web-agent.github.io/. We also include more details and screenshots of the web environment in the Supplementary Material.

4 The EMBODIED WEB AGENTS Benchmark Construction

In this section, we describe how we construct our EMBODIED WEB AGENTS benchmark. We will cover 5 domains of tasks: Navigation, Shopping, Traveling, Cooking and Geolocation. We show examples of the tasks in Figure 1, and a full pipeline of completing a task in Figure 2. Figure 3 summarizes the required level of each capability for successful task completion across domains, and Figure 4 shows which environments are utilized in different tasks.

Navigation Building upon the Outdoor Environment described in § 3.1, our navigation tasks evaluate an agent's spatial reasoning ability to reach destinations based on web-sourced directions. We use the OpenStreetMap website in § 3.3 to ensure reproducibility and consistent web interaction. To create diverse navigation scenarios, we prompt GPT-4o-mini to generate geographic coordinates across the aforementioned cities. These coordinates serve as either the start or end points of a task, and the graph structure centered around each point can be developed using our outdoor environment. During the prompting process, we also generate initial task instructions tied to the obtained coordinates. After identifying start or end points, we locate the corresponding counterparts using node adjacency relationships in the outdoor graph, forming a path within the environment. For evaluation purposes, we compute the shortest path using Dijkstra's algorithm as our ground-truth trajectory.

Navigation tasks require bidirectional interaction between web and embodied domains. The agent must input origin and destination into the map website to obtain directions, then ground these instructions in the embodied environment through turning actions and movements. Our benchmark includes 144 navigation tasks, each requiring both web interaction and embodied navigation. Since VLM-generated locations may have connectivity issues or misalignments with actual map data, we conduct human verification for all tasks to ensure their correctness and validity.

Shopping In real life, when buying products, we typically compare prices online, decide where to purchase based on pricing and store location information, place an order online, and then visit a physical store for pickup. Our shopping tasks evaluate the agent's ability to handle both online shopping and embodied environment interactions. The agent must place orders through our self-hosted shopping website dicussed in § 3.3, obtain store locations, and navigate in the outdoor environment to the correct store for pickup using the directions by OpenStreetMap; alternatively, it may also first navigate to a store and then place the order online.

In our benchmark, we simulate four stores located in distinct areas of Manhattan, New York. Our website lists a variety of items with product names, images, prices, and store information including distance and store name. The agent needs to *weigh both the price of the item and the store's location to make an optimal decision*, ultimately grounding web information into the embodied environment and navigating to the store for the selected item. To generate diverse scenarios, we design multiple templates with different items and user intents, which are listed in detail in our Supplementary Material. We also test the agent's ability to retrieve information across multiple browser tabs—*e.g.*, requiring the agent to complete a purchase, return to the homepage, switch to a map website, and

search for directions before embodied navigation. Some complex tasks require multiple rounds of web interaction and physical navigation within a single shopping scenario, testing agents' multi-source integration and sequential planning abilities. In total, our dataset contains 216 shopping tasks.

Traveling Inspired by how people consult web resources while traveling to navigate the physical world more effectively, we include traveling as a primary benchmark task. Using our custom-built outdoor environment and a pipeline similar to navigation tasks, we prompt a VLM to generate starting points, destinations, and initial task instructions, which we then refine into detailed, context-appropriate versions. Unlike pure navigation tasks that focus on following map directions and resolving map-reality inconsistencies, traveling tasks emphasize richer interaction between web resources and the embodied environment. For instance, when an agent encounters a significant landmark during navigation, as shown in Figure 1 (a) when it runs into a Gothic building, it may query Wikipedia to retrieve relevant information about that location. The agent is also expected to explore different architectural styles or historical landmarks, and ground Wikipedia descriptions to physical observations (*e.g.,* grounding the text descriptions of appearances of a Gothic building to the actual observation of the building). Web interactions in traveling tasks extend beyond map reading to include diverse informative sites, creating scenarios with multiple intertwined interactions between digital and physical domains. Our benchmark includes 110 traveling tasks, each requiring fluid movement between embodied navigation and web-based information retrieval.

Cooking As described in § 3.2, we use AI2-THOR as our indoor environment. To generate embodied cooking tasks for execution, we begin by identifying all ingredients available in the AI2-THOR kitchen scenes. We then manually search online for recipes that include these ingredients. Since online recipes are often noisy and may not align with the constraints of the AI2-THOR environment, we use Claude to refine them. Claude is guided by a predefined set of allowable agent actions in AI2-THOR environment to ensure the resulting recipes are executable. To increase task difficulty, we introduce confounders for most of the recipes by including pairs of recipes with the same name but differing in difficulty level, dietary type, ingredients used, or required cooking equipment. The users can filter out recipes based on these constraints by filter bars below the search bar (as in our self-hosted websites discussed in § 3.3). The next step is to curate a set of tasks based on collected recipes. For each scene, we retrieve recipes that match the available ingredients. The task instruction asks the agent to cook the corresponding dish. When a confounder exists for a given recipe, we introduce additional constraints — *e.g.*, "Diet type is vegetarian," "Use a tomato," — to disambiguate between recipe variants. If an ingredient does not exist in the scene, the agent is expected to go online to shop for it. The cooking tasks evaluate the agent's capability to perform long-trajectory planning in the indoor environment, and continuously check if the states match with the web instruction in the process. Our benchmark contains in total 911 cooking tasks. An exemplar task is in Figure 1 (b).

Geolocation Geolocation is a classic computer vision task Hays and Efros [2008], where models predict geographic coordinates of given images. Instead of treating it purely as a conventional vision problem, we reinterpret it based on its inherent characteristics as an embodied geolocation task. Inspired by the design of GeoGuessr, we move away from the single-image input setting and treat the model as an agent situated in an embodied environment. The agent is allowed to explore the outdoor environment we construct and ultimately output its estimated location. During exploration, the agent interprets storefront texts, visual cues, and street-view observations while accessing web information when needed to supplement its observations. The agent explores these environments freely, performing web interactions when additional information is needed. The task concludes when the agent has either 1) explored all possible positions or 2) collected sufficient information to confidently predict its location. This framework unifies embodied navigation, web-based reasoning, and visual grounding into a cohesive geolocation task. Our data collection is adapted from Huang et al. [2025], focusing on examples from existing geolocation datasets where models typically fail. We select coordinates where we hypothesize web information may improve prediction accuracy, then construct environments centered on these points using Google API. Geolocation evaluates the visual grounding ability of agents. An example is shown in Figure 1 (c). We collect 142 such data.

5 Experiments

In this section, we first introduce baseline LLM agents (§ 5.1) and evaluation metrics (§ 5.2) we use for experiments. We then perform result analysis (§ 5.3) on our EMBODIED WEB AGENTS benchmark. We group the results of Navigation, Shopping and Traveling together as they are all

related to outdoor planning. Please refer to the Supplementary Material for more experimental results, experimental setup, LLM prompts, qualitative examples and error cases, as well as more analyses.

5.1 Baseline LLM Agents

We evaluate four LLMs as our baseline agents: GPT-4o-mini, Gemini 2.0 Flash, Qwen-VL-Plus, and InternVL-2.5-latest. GPT-4o-mini is OpenAI's state-of-the-art multimodal model with strong performance in visual reasoning and real-time interaction. Gemini 2.0 Flash, by Google DeepMind, is optimized for speed and efficiency while maintaining robust vision-language capabilities. Qwen-VL-Plus, from Alibaba's Qwen Team, offers fine-grained image-text understanding. InternVL-2.5-latest, developed by Shanghai AI Lab, excels in spatial and semantic reasoning.

5.2 Evaluation Metrics

To comprehensively assess agent performance across physical and digital domains, we employ four evaluation metrics for outdoor planning and cooking: **Overall Accuracy** measures the success of complete cross-domain task execution, requiring both successful web task completion (reaching the terminal web state) and fulfillment in the embodied environment, representing holistic task completion that necessitates seamless integration of both domains; **Web-only Accuracy** evaluates the ability to successfully complete the web portion of a task, such as reaching the final step of a recipe, isolating digital domain independent of physical execution; **Embodied-only Accuracy** assesses an agent's ability to achieve all required physical state conditions in the embodied environment, such as properly slicing ingredients, or navigating to a desired place, measuring physical domain proficiency; and **Overall Completion Rate** represents the proportion of task progress achieved, indicating how much of the required state conditions have been fulfilled relative to the total task objectives.

5.3 Result Analysis

Task / Metric			GPT	Gemini	Qwen	Intern	Human
Outdoor Tasks	Navigation	Overall Accuracy	34.72	30.56	15.97	13.19	90.28
		Overall Completion Rate	52.08	48.96	36.81	26.04	91.32
		Web-only Accuracy	69.44	67.36	57.64	38.89	92.36
		Embodied-only Accuracy	48.61	46.53	31.25	23.61	90.97
	Shopping	Overall Accuracy	25.46	23.61	13.89	10.65	92.59
		Overall Completion Rate	31.94	30.56	18.52	14.35	93.52
		Web-only Accuracy	39.35	37.50	23.15	17.13	93.06
		Embodied-only Accuracy	34.26	32.41	17.59	12.96	93.98
	Traveling	Overall Accuracy	30.91	25.45	11.82	9.09	91.82
		Overall Completion Rate	50.91	48.18	34.55	20.91	93.64
		Web-only Accuracy	57.27	53.64	41.82	25.45	94.55
		Embodied-only Accuracy	47.27	44.55	29.09	19.09	92.73

Table 2: **Model Performance Across Different Outdoor Tasks.** There is a huge performance gap between LLM agents' performances and human performances.

Metric	Vision				Text				Human
	GPT	Gemini	Qwen	Intern	GPT	Gemini	Qwen	Intern	
Overall Acc	5.4	4.1	0.6	0.0	6.4	5.8	1.5	0.4	77.08
Completion Rate	40.26	35.62	15.91	9.73	39.16	38.92	17.20	10.02	85.37
Web Acc	59.71	47.74	28.65	10.64	57.08	62.23	35.89	15.58	100
Embodied Acc	8.7	6.1	2.2	0.9	10.5	8.2	4.1	1.3	77.08

Table 3: **Model Performance for Cooking Task.** The models achieve inferior overall accuracies.

Outdoor Planning For outdoor planning, we use GPT-4o-mini alongside Gemini 2.0 Flash, Qwen-VL-Plus, and InternVL-2.5-latest to evaluate performance across navigation, shopping, and traveling tasks (Table 2). For web observation, we follow the setting of VisualWebArena. We observe that: 1) GPT-4o-mini consistently leads across all metrics, with the highest accuracy in navigation (34.72%), shopping (25.46%), and traveling (30.91%), though still well below human performance. Gemini follows closely behind, while Qwen and Intern lag behind. 2) Web-only accuracy exceeds

embodied-only accuracy for all outdoor tasks, suggesting models handle digital information more effectively than physical navigation. 3) Generally, completion rates are satisfactory, while overall accuracies are very low across all tasks. This indicates models can execute parts of complex tasks but struggle with consistent cross-domain reasoning over longer sequences. 4) From task perspective, shopping and traveling involve richer interactions between the embodied environment and the web than navigation, and each task spans longer steps. As a result, the overall accuracy for shopping and traveling is noticeably lower than for navigation. This highlights the difficulty of cross-environment tasks, particularly those that are lengthy and involve multiple steps, for current models.

Cooking For cooking, we implement two distinct approaches: vision-based and text-based. Our vision-based implementation draws inspiration from VisualWebArena, utilizing screenshot images of websites enhanced with Set-of-Marks (SoM) annotations that highlight interactive elements. For embodied observations, we provide first-person visual perspectives from the agent's viewpoint within the AI2-THOR environment. The text-based implementation follows WebArena's methodology, representing web content through accessibility trees that capture the semantic structure of websites in textual form. For embodied observations, we extract structured scene graphs directly from AI2-THOR, providing explicit object relationships and states. We use Qwen-PLUS and InternLM-latest for Qwen and Intern models without vision.

Table 3 presents performance metrics for various models on the cooking task, comparing vision-based and text-based approaches against human performance. A substantial performance gap exists between AI models and humans, with the best model (text-based GPT-4o-mini) achieving only 6.4% overall accuracy compared to humans' 77.08%. Text-based models using structured scene graphs consistently outperform their vision-based counterparts using first-person views, suggesting current models struggle to ground visual observations effectively in cooking contexts. GPT-4o-mini and Gemini-2.0-Flash demonstrate substantially stronger performance than Qwen-VL-Plus/Qwen-PLUS and InternVL/InternLM across both modalities. Notably, similar to outdoor performances, all models perform significantly better on web-only tasks compared to embodied-only tasks, revealing that while current models can navigate recipe websites effectively, they struggle with physical execution requiring object manipulation and state tracking. Despite low overall accuracy, models achieve moderate completion rates, indicating partial task success but failure in full cross-domain integration.

Geolocation For geolocation tasks, we benchmark against FairLocator [Huang et al., 2025], a study analyzing VLM performance on GeoGuessr using Google Street View images. As shown in Table 4, the embodied web agent, capable of active exploration and web information access, significantly outperforms the passive baseline, particularly in identifying finer-grained locations like cities and streets. We observe consistent improvements across all models when moving from the baseline to embodied setting, suggesting the performance gains are model-agnostic. Interestingly, we also find that even when the retrieved Wikipedia search results are noisy or uninformative, the act of querying itself often helps the agent reason more confidently. This indicates that formulating search queries may serve as a form of self-supervision. This substantial improvement underscores the potential of integrating embodied and web domains to enhance performance across numerous real-world tasks, warranting further investigation.

	Setting / Model	Continent	Country	City	Street	All
Geolocation	**FairLocator**					
	GPT-4o-mini	90.85	81.69	73.24	1.41	1.41
	Gemini-2.0-Flash	93.66	85.92	78.17	0.70	0.70
	Qwen-VL-Plus	76.06	58.45	45.07	0.70	0.00
	InternVL2.5-Latest	77.46	62.68	52.11	1.41	1.41
	Embodied Web Agent					
	GPT-4o-mini	97.18	90.85	85.21	3.52	3.52
	Gemini-2.0-Flash	97.18	94.37	85.21	4.23	4.23
	Qwen-VL-Plus	80.28	69.01	49.30	0.00	0.00
	InternVL2.5-Latest	93.62	77.30	57.45	2.13	1.42

Table 4: **Model performance for geolocation task.** All models performed much better when predicting after interactively exploring the environment and querying the web than just using static images.

5.4 Error Analysis

Figure 5 presents a detailed breakdown of error types and their percentages that contribute to task failures in cooking tasks when using GPT-4o. Our analysis reveals that the primary challenges in embodied web agents lie not in isolated capabilities, but in their integration. While embodied errors (14.6%) and web errors (8.0%) occur, cross-domain errors (66.6%) overwhelmingly dominate the failure landscape — confirming that the critical bottleneck emerges at the intersection where physical and digital domains meet. The most prevalent failure pattern involves agents becoming trapped in single-domain cycles. In 23.6% of failures, agents get stuck in the embodied environment, repeatedly executing irrelevant physical actions without returning to the web for the next step. Similarly, in 13.2% of cases, agents remain fixed in web environments, endlessly clicking "next" through recipe pages without initiating cooking actions. In addition, agents often switch between environments without meaningful action (16.7%) or suffer from instruction-action misalignments (11.8%), such as slicing lettuce when a recipe instructs "slice the apple". Web interaction failures manifest as agents getting stuck in page loops (3.1%) or performing identical actions repeatedly (4.3%). In the embodied domain, agents fail to navigate to interactable objects (5.2%) or execute repeated actions (4.5%). These isolated domain errors are far less frequent than cross-domain integration failures, explaining why LLM agents achieve only 6.4% overall accuracy despite moderate performance on single-domain tasks. This confirms that embodied web agency presents unique challenges requiring focused research on mechanisms that bridge physical and digital reasoning.

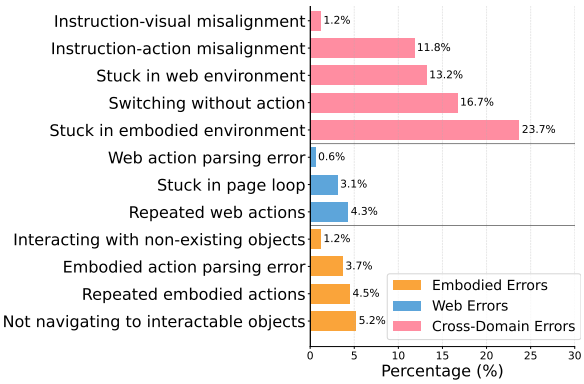

Figure 5: **Error Analysis for Cooking Tasks.** We can see that the majority of errors are cross-domain errors.

6 Conclusion

In this paper, we introduced EMBODIED WEB AGENTS, a new paradigm for AI research that bridges the artificial divide between physical and digital intelligence. Through our comprehensive benchmark spanning cooking, navigation, shopping, tourism, and geolocation tasks, we demonstrate that current AI systems face significant challenges in fluidly integrating embodied perception with web-based information retrieval. These findings establish a foundation for future research in integrated intelligence systems, highlighting the need for developing AI agents that can seamlessly traverse physical and digital worlds. A limitation is our reliance on simulated agents, which may not fully capture the complexity and unpredictability of physical-digital interactions of real robots.

Broader Impact

Our EMBODIED WEB AGENTS research presents both opportunities and challenges for society. On the positive side, agents that bridge physical and digital domains could enhance accessibility for individuals with mobility limitations, support contextualized learning environments, and improve emergency response through integrated information access. However, several risks warrant attention. First, these agents may exhibit "dual-domain hallucination," where errors propagate across physical and digital realms, compounding misinformation. Second, systems that connect physical environments with web platforms introduce novel privacy concerns beyond those in either domain alone.

To mitigate these concerns, our benchmark provides transparent evaluation protocols that can identify cross-domain errors. We designed our environments as simulations that don't interact with real-world systems, limiting immediate risks while providing valuable research insights. By releasing our benchmark to the research community, we aim to encourage the development of more robust embodied web agents with improved error detection mechanisms before deployment in real-world settings.

Acknowledgment

We thank anonymous reviewers for their helpful comments. This work was partially supported by U.S. DARPA ECOLE Program No. #HR00112390060, ONR grant N00014-23-1-2780, DARPA ANSR program FA8750- 23-2-0004, Amazon, Google and Apple Research Award. Chang was supported in part by a grant from DARPA to the Simons Institute for the Theory of Computing.

References

Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2019. doi: 10.1109/cvpr.2019.01282. URL http://dx.doi.org/10.1109/CVPR.2019.01282. 4

Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Sam Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. *Advances in Neural Information Processing Systems*, 36:28091–28114, 2023. 1, 3

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model, 2023. 4

Alexandre Drouin, Maxime Gasse, Massimo Caccia, Issam H Laradji, Manuel Del Verme, Tom Marty, Léo Boisvert, Megh Thakkar, Quentin Cappart, David Vazquez, et al. Workarena: How capable are web agents at solving common knowledge work tasks? *arXiv preprint arXiv:2403.07718*, 2024. 3

Ruofei Du and Amitabh Varshney. Social street view: Blending immersive street views with geo-tagged social media. In *Proceedings of the 21st International Conference on Web3D Technology*, pages 77–85. ACM, 2016. doi: 10.1145/2945292.2945297. URL http://www.socialstreetview.com. 4

Ruofei Du, David Li, and Amitabh Varshney. Geollery: A mixed reality social media platform. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*, pages 1–13. ACM, 2019. ISBN 978-1-4503-5970-2. doi: 10.1145/3290605.3300915. URL https://doi.org/10.1145/3290605.3300915. 4

GeoGuessr. URL https://www.geoguessr.com/. 7

Lukas Haas, Michal Skreta, Silas Alberti, and Chelsea Finn. Pigeon: Predicting image geolocations, 2023. 4

James Hays and Alexei A Efros. Im2gps: estimating geographic information from a single image. In *2008 ieee conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008. 7

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. WebVoyager: Building an end-to-end web agent with large multimodal models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6864–6890, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.371. URL https://aclanthology.org/2024.acl-long.371/. 3, 4

Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *NeurIPS*, 2023. 4

Hexiang Hu, Yi Luan, Yang Chen, Urvashi Khandelwal, Mandar Joshi, Kenton Lee, Kristina Toutanova, and Ming-Wei Chang. Open-domain visual entity recognition: Towards recognizing millions of wikipedia entities, 2023. URL https://arxiv.org/abs/2302.11154. 3

Jingyuan Huang, Jen-tse Huang, Ziyi Liu, Xiaoyuan Liu, Wenxuan Wang, and Jieyu Zhao. Vlms as geoguessr masters: Exceptional performance, hidden biases, and privacy risks. *arXiv preprint arXiv:2502.11163*, 2025. 7, 9

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov, and Daniel Fried. VisualWebArena: Evaluating multimodal agents on realistic visual web tasks. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 881–905, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.50. URL https://aclanthology.org/2024.acl-long.50/. 1, 3

Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. AI2-THOR: An Interactive 3D Environment for Visual AI. *arXiv*, 2017. 3, 5

Chengshu Li, Fei Xia, Roberto Martín-Martín, Michael Lingelbach, Sanjana Srivastava, Bokui Shen, Kent Elliott Vainio, Cem Gokmen, Gokul Dharan, Tanish Jain, Andrey Kurenkov, Karen Liu, Hyowon Gweon, Jiajun Wu, Li Fei-Fei, and Silvio Savarese. igibson 2.0: Object-centric simulation for robot learning of everyday household tasks. In Aleksandra Faust, David Hsu, and Gerhard Neumann, editors, *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pages 455–465. PMLR, 08–11 Nov 2022. URL https://proceedings.mlr.press/v164/li22b.html. 3

Manling Li, Shiyu Zhao, Qineng Wang, Kangrui Wang, Yu Zhou, Sanjana Srivastava, Cem Gokmen, Tony Lee, Li Erran Li, Ruohan Zhang, et al. Embodied agent interface: Benchmarking llms for embodied decision making. In *NeurIPS 2024*, 2024. 3

Kevin Qinghong Lin, Linjie Li, Difei Gao, Zhengyuan Yang, Shiwei Wu, Zechen Bai, Weixian Lei, Lijuan Wang, and Mike Zheng Shou. Showui: One vision-language-action model for gui visual agent, 2024. 4

Evan Zheran Liu, Kelvin Guu, Panupong Pasupat, Tianlin Shi, and Percy Liang. Reinforcement learning on web interfaces using workflow-guided exploration. In *International Conference on Learning Representations (ICLR)*, 2018. URL https://arxiv.org/abs/1802.08802. 3

Xing Han Lù, Zdeněk Kasner, and Siva Reddy. Weblinx: Real-world website navigation with multi-turn dialogue. In *Forty-first International Conference on Machine Learning (ICML)*, 2024. 3

Manolis Savva*, Abhishek Kadian*, Oleksandr Maksymets*, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 3

Harsh Mehta, Yoav Artzi, Jason Baldridge, Eugene Ie, and Piotr Mirowski. Retouchdown: Releasing touchdown on StreetLearn as a public resource for language grounding tasks in street view. In Parisa Kordjamshidi, Archna Bhatia, Malihe Alikhani, Jason Baldridge, Mohit Bansal, and Marie-Francine Moens, editors, *Proceedings of the Third International Workshop on Spatial Language Understanding*, pages 56–62, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.splu-1.7. URL https://aclanthology.org/2020.splu-1.7/. 4

Piotr Mirowski, Matt Grimes, Mateusz Malinowski, Karl Moritz Hermann, Keith Anderson, Denis Teplyashin, Karen Simonyan, koray kavukcuoglu, Andrew Zisserman, and Raia Hadsell. Learning to navigate in cities without a map. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/e034fb6b66aacc1d48f445ddfb08da98-Paper.pdf. 4

Tzuf Paz-Argaman and Reut Tsarfaty. RUN through the streets: A new dataset and baseline models for realistic urban navigation. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6449–6455, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1681. URL https://aclanthology.org/D19-1681/. 4

Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. A generalist agent, 2022. 4

Bokui Shen, Fei Xia, Chengshu Li, Roberto Martín-Martín, Linxi Fan, Guanzhi Wang, Claudia Pérez-D'Arpino, Shyamal Buch, Sanjana Srivastava, Lyne P. Tchapmi, Micael E. Tchapmi, Kent Vainio, Josiah Wong, Li Fei-Fei, and Silvio Savarese. igibson 1.0: a simulation environment for interactive tasks in large realistic scenes. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, page accepted. IEEE, 2021. 3

Tianlin Shi, Andrej Karpathy, Linxi Fan, Jonathan Hernandez, and Percy Liang. World of bits: An open-domain platform for web-based agents. In *International Conference on Machine Learning*, pages 3135–3144. PMLR, 2017. 1, 3

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. ALFRED: A Benchmark for Interpreting Grounded Instructions for Everyday Tasks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. URL https://arxiv.org/abs/1912.01734. 3

Sanjana Srivastava, Chengshu Li, Michael Lingelbach, Roberto Martín-Martín, Fei Xia, Kent Vainio, Zheng Lian, Cem Gokmen, Shyamal Buch, C. Karen Liu, Silvio Savarese, Hyowon Gweon, Jiajun Wu, and Li Fei-Fei. Behavior: Benchmark for everyday household activities in virtual, interactive, and ecological environments, 2021. 3

Shuo Xing, Zezhou Sun, Shuangyu Xie, Kaiyuan Chen, Yanjia Huang, Yuping Wang, Jiachen Li, Dezhen Song, and Zhengzhong Tu. Can large vision language models read maps like a human?, 2025. 4

Jihan Yang, Runyu Ding, Ellis Brown, Xiaojuan Qi, and Saining Xie. V-irl: Grounding virtual intelligence in real life, 2024. URL https://arxiv.org/abs/2402.03310. 4

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757, 2022. 1, 3

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Yonatan Bisk, Daniel Fried, Uri Alon, et al. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854*, 2023. URL https://webarena.dev. 1, 3, 4, 6

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer:[Yes]

   Justification: We clearly state our main claims in abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations in Conclusion.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA] .

Justification: This paper doesn't introduce new theorems.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Yes, we fully disclose all the information in our Experiments sections as well as in the Supplementary Material

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the ssame dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide the whole dataset and codes in the submitted supplemental material. We will release our code and data publicly as well.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes, we specify them in the Experiments section as well as in the Supplementary Material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Yes, we report the average accuracies across multiple runs.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [NA]

   Justification: We experiment with LLM APIs. Thus, no local compute / memory recorded.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Yes, it conforms with the Code of Ethics

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Yes, we discuss them in Broader Impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Yes, we credit them in the references. We mention the licenses in the Supplementary Material.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide detailed documentation in the Supplementary Material.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: We add the instructions for acquiring human performance in the Supplementary Material.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: We did not research with human subjects. We only acquired human performance for comparison.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: We discussed LLM usage in the Baselines LLM Agents subsection of the Experiments section, as well as in the Supplementary Material.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.