Text-to-Scenario Generation over Semantic Worlds with Controllable Human Behaviors

Anonymous Authors

***Abstract*—Social navigation is critical for robots in human environments, yet most simulators evaluate methods in static or oversimplified settings. We propose a text-to-dynamic scenario generator that compiles natural-language prompts into interactive simulation worlds with time-evolving environments and controllable human behaviors. Prompts are transformed into parameterized scenario programs specifying layouts, roles, intents, motion styles, interaction goals, temporal schedules, and exogenous events. Human activity is instantiated via probabilistic, time-varying parameters (e.g., enter → queue → order → sit/move) using behavior trees coupled using LLM-based reasoning and a curated database of behavior models. These behaviors are grounded in semantically annotated zones and Regions of interest (ROIs) (e.g., doors, counters, seats), constraining where agents move and which actions they can perform for realistic interaction. To ensure simulator independence, we introduce a unified pedestrian-state wrapper that abstracts over commonly used simulators. Using this approach, we generate realistic social scenarios directly integrable into Gazebo, Isaac Sim, and Unity, and evaluate their realism and usability through commonly utilized metrics and user studies.**

I. Introduction

Robots operating in human environments must reason not only about static geometry but also about how people move, interact, and change intentions over time. However, much of today's evaluation in social navigation is still performed in static or overly stylized settings. When dynamics are modeled, they are often simulator-specific or simplified, which limits realism and reproducibility. Although photorealistic environments increasingly provide high visual fidelity, they still lack semantic social structure for human modeling, preventing behaviors such as queuing, gathering, or waiting from being meaningfully grounded in the scene. To address these limitations, we introduce a ROS 2 framework for reproducible, text-to-dynamic scenario generation that takes prompt inputs and generates interactive 3D worlds and corresponding scenarios. Thereby, we integrate semantic world representations, standardized pedestrian state definitions, and generative modules for scenario authoring. The key contributions are:

- A complete end-to-end pipeline that transforms natural-language inputs into executable interactive simulation worlds and scenarios. Unlike prior approaches that treat environment layout and crowd behavior as decoupled stages, our pipeline grounds agent roles, intents, and routines directly in semantically meaningful regions and objects (ROIs/POIs) of the generated world. By combining lightweight RAG, large language models, and diffusion-based crowd priors, the system produces contextually coherent behaviors such as queuing, waiting, gathering, and evacuation that are intrinsically linked to the spatial and semantic structure of the environment.
- Two core abstractions that enable semantic grounding, portability, and reproducibility across simulators: (i) a human- and machine-editable semantic world representation (`world.yaml`) that encodes structural layout, interactive affordances, and explicit ROIs/POIs for behavior anchoring; and (ii) a unified ROS 2 pedestrian state representation (`socsim_people_msgs`).
- A full integration of the proposed pipeline into standard robotics workflows, including native ROS 2 interfaces and direct integration into the RViz task panel that enables on-the-fly world and scenario generation, modification, and execution. Scenarios can be instantiated, regenerated, and benchmarked interactively within the same visualization and control environment within ROS2.

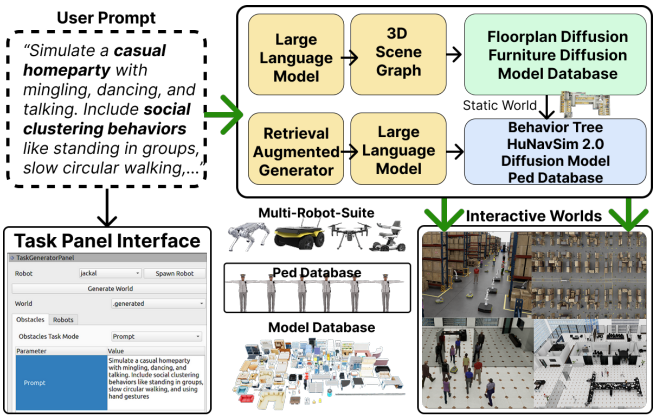

Fig. 1: An overview of our system pipeline. User prompts and parameters are entered through a task panel and parsed using LLMs, a RAG system, and collected databases. The system compiles the corresponding assets and behaviors into interactive scenarios that can be loaded into different simulators (e.g., Isaac Sim, Gazebo, Unity). This enables scalable generation of diverse scenarios, including edge cases, for training, testing, and benchmarking of navigation approaches.

We evaluate the platform by generating a variety of unseen scenarios across different environments and comparing them against state of the art baselines on commonly used metrics and a user study on usability and perceived realism. The results indicate that the proposed framework substantially lowers the barrier for generating complex, semantically rich evaluation settings such as edge cases, which can accelerate

the development and testing of navigation approaches for robotics.

II. Related Works

With recent advances in computer vision, generative modeling, and simulation fidelity, world and scenario synthesis has become an essential part of the development cycle across several industries such as autonomous driving and robotics.

Recent advances in 3D/4D reconstruction and generative scene modeling including NeRF-style radiance fields [1], Gaussian Splatting[2], [3], [4], diffusion-based scene synthesis [5], [6], [7], or procedural techniques [8], [9], produce richly detailed environments. However, these pipelines largely neglect human activity: in most simulators, people are added post hoc via scripts, leaving a gap between photorealistic geometry and realistic, interactive behavior. On the other hand, most current navigational evaluation tools, such as HuNavSim [10], RoboBench [11], and SocNavBench [12], focus primarily on the fidelity of human simulation rather than generating specific scenarios that test a robot's navigation and reactive capabilities. Recent works such as SEAN 2.0 [13] and Arena 5.0 [14] are exceptions that seek to expose robots to various social situations. However, these works provide only a limited set of scenarios and environments, offering little control over the scenario itself.

Another long-standing problem is realistic simulation of pedestrian behavior. Current works typically synthesize pedestrians with simplified models such as the *Social Force Model* (SFM) or (Reciprocal) *Velocity Obstacles* (ORCA/RVO) [15], [16], often via toolkits like `pedsim_ros` (based on [17]) or MengeROS [18]. These are efficient but local-rule based, missing higher-order phenomena (stop-and-talk, group cohesion, intent-driven queuing). More recent systems—*HuNavSim* and *HuNavSim 2.0*, increase realism with richer profiles and interactions in ROS/Gazebo [19], [20], yet adoption remains limited due to simulator-specific integration and scenario authoring that still relies on handcrafted or random seeds.

Meanwhile, advances in the computer vision and graphics communities enable *single-human* motion/intent modeling and scene-conditioned animation [21], [22], [23], [24]. LLM-based motion generation frameworks such as [25], [26] explore language-conditioned motion and scaling laws. Recent works also propose LLM reasoning for navigating pedestrian agents in cluttered scenes [27], [28], [29]. Yet these works typically output actions and behavior patterns for a single actor and stop short of robotics-ready, *multi-agent* crowds embedded into ready-to-use simulation scenarios for robotics development.

Recent works by Zhou et al. such as PedGen [30] or MetaUrban [31] derive or synthesize pedestrian dynamics from video or generative priors for outdoor navigation. For indoor human motion, *Trace&Pace* uses diffusion for realistic trajectories [32]; however, turning such pipelines into *interactive, simulator-ready* assets (e.g., Isaac Sim) remains non-trivial. Most related to our work are those by Ji et al. [33], which compiles LLM-parsed text into crowd animations using text-guided diffusion for spatial distributions with RVO control. Another similar work utilizes concurrent LLM+VGAE pipelines targeting text/graph-to-crowd synthesis [34]. However, these methods are not or only partly released, and primarily target simulator-agnostic use cases such as CARLA for autonomous driving, making reproducibility and integration into the robotics community an open challenge.

We address these gaps with a robotics-focused pipeline that generates interactive 3D scenarios from text prompts, couples semantic world specifications with LLM reasoning, and provides a unified pedestrian-state interface across simulators. Unlike prior tools, it emphasizes socially salient situations (e.g., emergencies, queues, blocked doors) by binding human routines to world semantics rather than relying on generic motion patterns that ignore context.

III. Methodology

A. Overview

Our approach provides an end-to-end pipeline that transforms natural-language prompts into semantically grounded multi-agent scenarios for social navigation. Given a user prompt, our system first parses the text into structured scene and behavior representations, binds roles and intents to our proposed semantic world specification (`world.yaml`) with annotated ROIs and POIs, and instantiates behavior trees with probabilistic, time-varying parameters. In the next step, pedestrian trajectories are realized through a unified state interface (`socsim_people_msgs`), which decouples behavior generation from the underlying simulator and enables execution in Gazebo, Isaac Sim, or Unity. All scenario artifacts—including prompts, seeds, world specifications, behavior trees, and pedestrian-state logs—are stored as reusable *scenario cards*, supporting reproducibility, benchmarking, and training reuse. Fig. 2 illustrates the complete pipeline, which consists of two parts: 1) the World Generation Pipeline and 2) the Scenario Generation Pipeline.

B. World Generation Pipeline

Different from related works, which treat world generation and crowd generation as separate tasks, we introduce a number of methods to generate semantically annotated structural worlds designed to operate in close synergy with our scenario crowd generation pipeline. This requires a robust mechanism for producing structural layouts with meaningful regions and valid geometry. To meet this requirement, we first propose a world representation format that encodes structural layout, spatial semantics, and interactive affordances. Next, we implement a RAG pipeline that synthesizes floor plans from high-level scene graphs while ensuring compliance with domain-specific building standards. As a result, an abstract

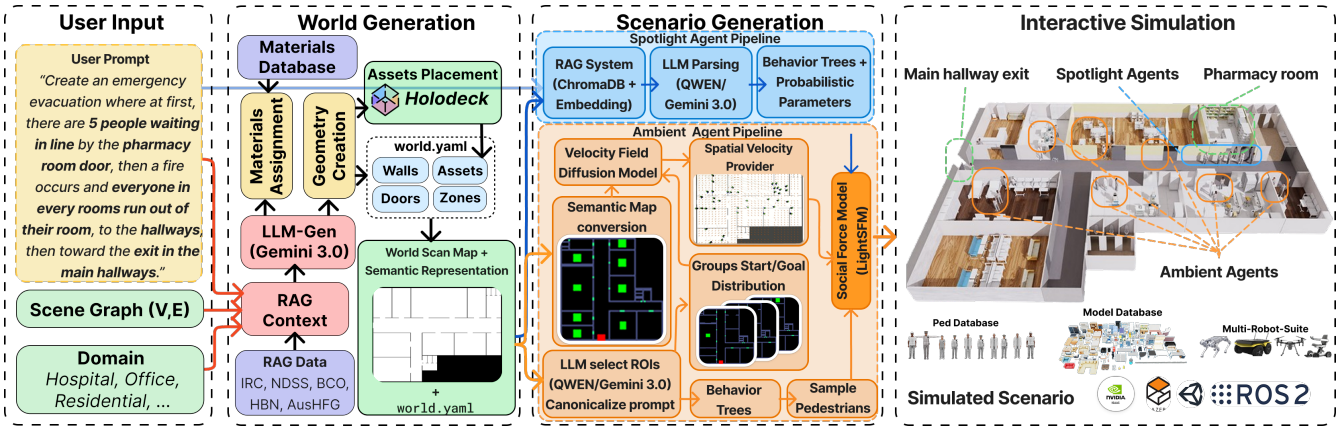

Fig. 2: Overview of the end-to-end *Prompt-to-Scenario Workflow* that transforms a natural language prompt into executable multi-agent simulation states. The pipeline consists of two modules, the world generation and scenario generation modules. The user inputs a scene graph encoding room nodes and adjacency edges, a domain classifier (Hospital, Residential, Office,...), and a user prompt. A RAG module retrieves dimensional constraints from authoritative building codes (IRC, NDSS, BCO, HBN, AusHFG, ...) based on the detected domain, injecting compliance standards into the LLM prompt. The LLM generates JSON polygon coordinates which are parsed into Shapely geometries. The output world specification (`world.yaml`) defines zones—semantically labeled regions that serve as Regions of Interest (ROIs) for behavior anchoring along with walls assigned materials and doors that encode navigable openings between adjacent zones. These zones/ROIs ground pedestrian behaviors by constraining feasible actions (e.g., wait in patient room, queue at reception) and are directly referenced in behavior tree specifications. The scenario generation pipeline consists of two parallel pipelines. The Spotlight Agent pipeline (top) uses retrieval-augmented generation (RAG) over a semantically annotated behavior tree library to parse the prompt into structured agent roles, parameters, and behavior trees. The Ambient Agent pipeline (bottom) extracts semantically grounded group specifications from the prompt and world specification output of the previous stage, generates crowd-level velocity fields via a diffusion model, and instantiates large populations of pedestrians guided by these fields. Both pipelines are integrated in the simulator through a social force model (lightSFM by default), producing coherent, scalable crowd behaviors consistent with the user intent and environment layout.

scene graph $G = (V, E)$—with vertices V denoting rooms and edges E encoding adjacency is transformed into a fully specified `WorldDescription` format.

1) Semantic World Format: An essential abstraction in our system is the representation of worlds. Existing simulators often rely on ad hoc formats or mesh-based geometries that lack machine-readable semantics, making it difficult to bind human behaviors to the environment in a context-aware manner. To address this, we propose a human- and machine-editable YAML specification (`world.yaml`) that encodes structural layout, spatial semantics, and interactive affordances. The format subdivides an environment into zones with explicit annotations of geometry, entry points, and functional roles, allowing behaviors to be bound not only to free space but also to meaningful regions and objects.

Each `world.yaml` file specifies editable structural elements such as walls (start, end, height), doors that act as explicit congestion sources, and floor properties for rendering and semantic segmentation. It further annotates ROIs such as queuing areas, seating areas, corridors and POIs, e.g., specific doors, tables, entrances with polygons and identifiers, and defines spawnable entities such as counters, tables, and seats. These annotations ensure that agent behaviors—such as "queue at the counter" or "sit in the seating area" are grounded in semantic context rather than generic coordinates. An exemplary `world.yaml` is given in Fig. 3.

```yaml
zones:
  - name: canteen_hall
    walls:
      - {start: [0.0, 0.0], end: [10.0, 0.0],
        height: 3.0, style: "white_paint"}
    doors:
      - {name: "entrance_east", pose: [9.8, 2.0],
        width: 1.2}
    floor: {material: "polished_concrete"}
    segmentation:
      rois:
        - {name: "queue_lane", polygon:
          [[2,1],[8,1],[8,2],[2,2]]}
        - {name: "seating_area", polygon:
          [[1,3],[9,3],[9,8],[1,8]]}
    entities:
      - {type: "counter", name: "food_counter",
        pose: [8.5, 1.5]}
      - {type: "table", name: "t_01", pose: [3.0,
        5.0]}
```

Fig. 3: `world.yaml` excerpt with zones, ROIs/POIs, and entities.

The semantic world format serves two purposes. First, it provides a compact and editable representation, enabling both manual editing and automatic synthesis. Second, it acts as the grounding layer for scenario generation pipeline: ROIs and POIs referenced in user prompts are resolved to entries in `world.yaml`, ensuring that agent intents (e.g., "move to counter," "exit through east door") are semantically consistent with the environment. To increase realism, the format is compatible with both procedural generators and diffusion-based models that can synthesize or refine assets consistent with the YAML specification. This integration allows us to

rapidly prototype a large number of diverse environments.

C. ROI World Generation Using RAG System

To generate these semantic worlds, we propose a combined approach utilizing a RAG pipeline, LLMs, as well as open-source models such as Procthor [35] and Holodeck [36] as backends for asset placement. The pipeline comprises three stages: Preprocessing, RAG context injection, and Inference/-Postprocessing.

a) Preprocessing: The input scene graph is converted into a structured prompt for a LLM. By analyzing the user prompt, the system performs automatic building type classification. Hospital indicators (e.g., "patient room," "ICU") trigger retrieval of healthcare infrastructure standards, while office indicators (e.g., "meeting room," "conference") invoke workspace guidelines. The system defaults to residential standards when no domain-specific indicators are detected.

b) RAG Context Injection: Our main goal of this stage is to provide world generation with authorized, reliable constraints, reducing hallucinated dimensions, and enabling transparent attribution. Therefore, we maintain a curated *RAG knowledge base* that we build as part of our system. It encodes the room size standards along with provenance metadata (source, jurisdiction, confidence). Based on the classified building type, the system retrieves relevant constraints from the knowledge base, which are formatted as natural language and injected into the LLM prompt.

By default, the following standards are included in our knowledge base:

- *IRC 2021* [37]: Min. habitable room 70 ft^2
- *UK NDSS 2015* [38]: Bedroom mins. (7.5–11.5 m^2)
- *BCO Guide* [39]: Workspace density 10 m^2/person
- *HBN 03-01* [40]: Patient bedroom 15 m^2
- *AusHFG* [41]: ICU room 25 m^2

The knowledge base follows a simple hierarchical schema that supports both retrieval and attribution:

```
Reference
+-- ref_id, building_type, jurisdiction
+-- sources[]: URLs, section hints
+-- constraints[]
    +-- type: min|max|recommended
    +-- value, unit, confidence
```

c) Inference and Postprocessing: The augmented prompt is sent to a LLM (Gemini 3 Pro), which generates a JSON response with polygon coordinates for rooms and doors. The postprocessing stage parses this response, constructs Shapely [42] geometries, and assembles the final `WorldDescription`. Wall segments are derived by computing boundary differences between room polygons and door openings. Each zone receives material assignments based on room type conventions. Finally, the RAG pipeline outputs directly populates the *zones* array in our world format as a `world.yaml` file. Each zone includes: (1) geometry as corner positions, (2) walls with material specs, (3) doors with width/pose, and (4) semantic name (e.g., `patient_room_01`). This ensures ROIs/POIs in behavior specifications are grounded in geometrically valid, standards-compliant regions.

D. Scenario Generation Pipeline

The second part of our overall pipeline is the scenario generation pipeline populating realistic humans and crowds within the generated worlds. At the core of the system is a unified representation format for pedestrian states based on the ROS 2 message type `socsim_people_msgs` as well as a RAG system both of which will be explained in the following sections.

E. Pedestrian State Descriptions

In order to ensure reproducibility and stability across all simulators, we introduce a unified abstraction layer via the ROS 2 message type `socsim_people_msgs`. This message type is used internally for module-to-module communication and externally as the bridge to simulators, ensuring that behavior logic, high-level intent, and low-level kinematics can be expressed in a single, consistent format. The message schema is shown in Table I. In addition to standard state fields such as identifier, pose, and velocity, we explicitly encode semantic and social attributes required for realistic human-robot interaction scenarios. The `state` field tracks the current activity (e.g., walking, queuing, interacting), while `role` enables differentiation of agent types with specific priors (e.g., joggers vs. hospital staff). The `intent` field links an agent to explicit ROIs or POIs in the semantic world specification, enabling context-aware navigation such as moving toward a counter or entering a seating area, while`group_id` allows coupling of multiple agents into social groups, enabling phenomena such as cohesion, flocking, or group queuing. Proxemic information captures social spacing and compliance parameters, while a metadata field stores simulation artifacts such as random seeds or current behavior-tree nodes, which are critical for reproducibility and ablation studies.

TABLE I: Key fields in `socsim_people_msgs`.

Field	Description
`header`	ROS header (timestamp, frame).
`id`	Persistent pedestrian identifier.
`pose`	3D position + orientation (SE(3)).
`twist`	Linear / angular velocity.
`state`	{walking, standing, queuing, sitting, interacting, ...}.
`role`	{jogger, staff, stroller_pusher, ...}.
`intent`	Target ROI/POI, goal id, schedule hints.
`group_id`	Group membership for social coupling.
`proxemics`	Comfort radius / compliance parameters.
`metadata`	Key–value bag (seed, behavior_tree_node, tags).

Internally, the message type supports reasoning and logging of dynamic states, while externally it is consumed by the *Gazebo world plugin* and the *Isaac Sim bridge*, which translate the abstract specification into simulator-specific commands.

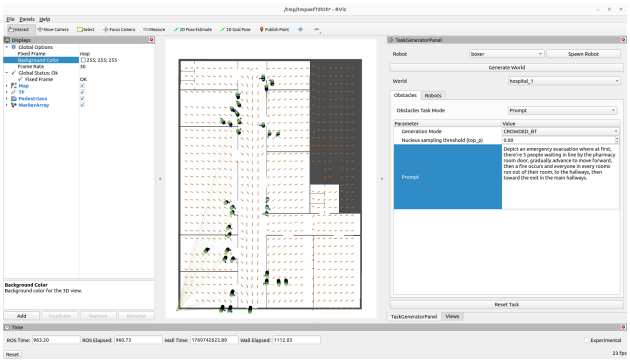

Fig. 4: Visualization of the desired force field derived from a diffusion-generated velocity field.

F. Pedestrian Force Modeling

LLM-generated behavior trees grounded in the Social Force Model (SFM) [15] and its empirical extensions [43], [44] enable semantically rich and expressive control of pedestrian motion, including collision avoidance and social interaction. However, such behavior-tree-based control does not scale to large pedestrian populations due to the limited context window of LLMs. In contrast, diffusion-based velocity field priors scale efficiently to dense crowds but lack explicit modeling of local interactions. To combine the strengths of both paradigms, we adopt the `LightSFM` framework[1], derived from SFM, and integrate velocity fields generated by diffusion models into its force-based formulation.

A key challenge in our system is incorporating image-based velocity fields, used to guide large populations of pedestrians, into force-based pedestrian dynamics. Directly applying velocity commands leads to frequent collisions and unstable behaviors, as the generated velocity fields have finite spatial resolution and do not explicitly encode local interaction constraints. In prior work on diffusion-based crowd generation [45], this issue is addressed using Reciprocal Velocity Obstacles (RVO) [46]. However, RVO operates purely at the velocity level and ignores social interaction, making it incompatible with force-based simulators such as LightSFM.

To resolve this mismatch, we introduce an integration strategy that converts generated velocity fields into socially compliant navigation behaviors. Specifically, we implement a custom behavior tree node, termed `FollowVelocityField`, which interprets the local velocity vector as a navigation intent. This node applies the Algorithm 1 to transform the sampled velocity into an intermediate goal location and a corresponding goal-attractive force, preserving the motion direction implied by the diffusion model. Additionally, the resulting goal-attractive force field can be visualized in RViz, providing an interpretable representation, as shown in Fig. 4. The resulting force is then combined with the standard LightSFM forces arising from nearby pedestrians and obstacles, allowing collision avoidance and social interaction to be handled consistently within the force-based framework.

[1]https://github.com/robotics-upo/lightsfm

In consequence, our approach not only avoids the need for external collision avoidance mechanisms, but also combines the strengths of diffusion-based crowd generation and social force modeling.

Algorithm 1: Velocity Field to Goal Force Conversion

Input: Current pedestrian position $pos \in \mathbb{R}^2$; world specification W (`world.yaml`); velocity field $V \in \mathbb{R}^{64\times64\times2}$; time step dt; goal tolerance tol.
Output: Goal-attractive force F_{goal}

$index \leftarrow \left(\frac{pos_0}{\text{world\_size}_0} \times 64,\ \frac{pos_1}{\text{world\_size}_1} \times 64 \right)$
`// Velocity grid index`
$vel \leftarrow V_{index}$
$goal \leftarrow pos + vel \times dt$
$d \leftarrow goal - pos$
if $\|vel\| > 0$ ***and*** $\|d\| < 2 \times tol$ **then**
  $goal \leftarrow pos + d \times \frac{2\times tol}{\|d\|}$
end
$F_{\text{goal}} \leftarrow \text{LIGHTSFMDESIREDFORCE}(goal, pos)$

This design naturally supports two classes of agents termed *Ambient Agents*, and *Spotlight Agents*. *Ambient Agents* rely primarily on the `FollowVelocityField` node to achieve scalable navigation guided by velocity field priors, while *Spotlight Agents* are controlled by richer behavior trees composed of semantically meaningful nodes that encode detailed actions. This design improves scalability compared to generating behavior trees for large populations solely through LLM inference, which is computationally expensive, error-prone, and constrained by limited context windows.

G. Scenario Generation Pipeline

Having described how individual and pedestrian motion is modeled and executed, we now present the end-to-end pipeline that maps high-level textual prompts into executable simulator states and crowd behaviors (scenario generation part in Fig. 2). The workflow consists of two parallel pipelines: one for generating detailed behaviors for *Spotlight Agents*, and another for synthesizing motion dynamics for large populations of *Ambient Agents*. These pipelines are described in detail below. Our system is embedded within RViz, where the user can set, generate and change the scenarios on the fly. Exemplary generations are depicted in Fig. 5. Table II lists scenario categories to prompt mappings.

a) Spotlight Agent Parsing via RAG and LLMs: We extend the HuNavSim behavior library [19] with enriched behavior tree nodes to support robust multi-agent interactions, such as structured queuing behaviors in which agents advance sequentially after timed intervals, as well as dynamic goal assignment and navigation. Each behavior node is annotated with semantic descriptions of its inputs, outputs, and execution logic, then converted into text chunks and embedded using *gemini-embedding-001* to form a database of behavior nodes. Given a prompt, a RAG module implemented with ChromaDB retrieves a set of candidate behavior nodes that

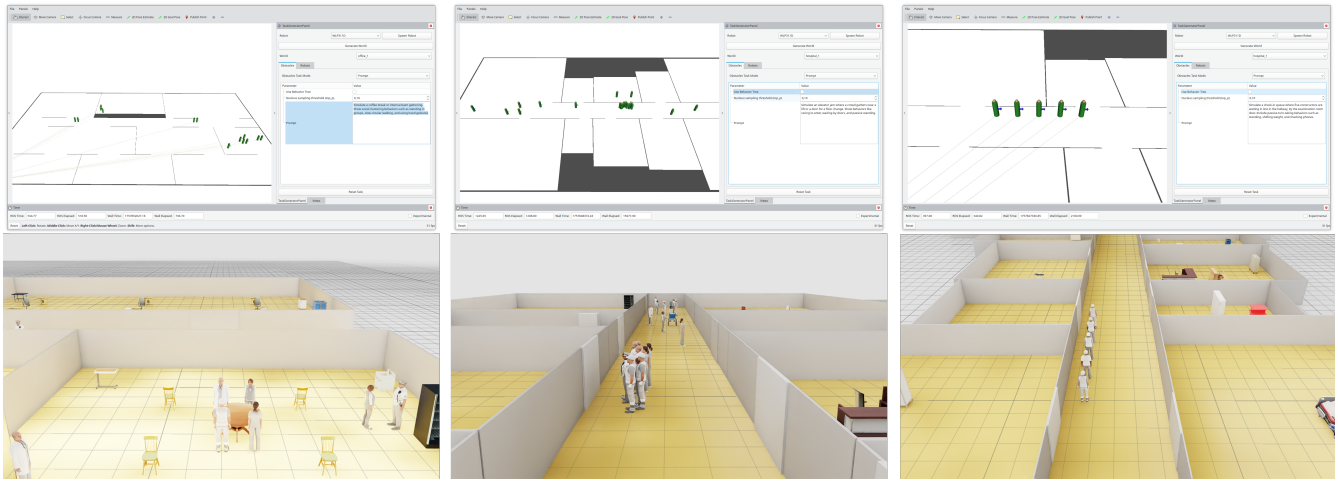

Fig. 5: Examples of three text prompts entered through the RViz task panel (top row) and their corresponding generated worlds in a hospital environment (bottom row). The prompts specify situations such as patients queuing at a reception desk, staff gathering for a brief exchange near a corridor intersection, and visitors waiting by an elevator. Each prompt is processed through the pipeline to instantiate pedestrians, assign behaviors, and anchor interactions to semantically meaningful regions of interest such as desks, doors, and waiting areas.

TABLE II: Scenario categories with POIs, ROIs, and behaviors (Bs). Keywords illustrate how prompts are mapped.

Scenario (Keywords)	**Details**
Emergency (*alarm, fire, evacuation*)	**POIs:** doors, exits, stretchers **ROIs:** corridors, assembly zones **Bs:** evacuation, panic running, staff rushing
Queuing (*queue, reception, elevator*)	**POIs:** counters, stalls, elevator doors **ROIs:** queue lanes, waiting zones **Bs:** enter, queue, idle, order, board elevator
Social / Home party (*party, mingle, gathering*)	**POIs:** tables, sofas, music systems **ROIs:** living rooms, social areas **Bs:** group talk, clustering, rejoining, dancing
Coffee Break (*coffee, chat, break*)	**POIs:** coffee machines, corridor nodes **ROIs:** break areas, hallways **Bs:** stop-and-chat, loiter, continue walking
Reception / Check-in (*reception, counter, registration*)	**POIs:** counters, desks, terminals **ROIs:** reception areas, queues **Bs:** approach, queue, register, depart
Bottleneck (*congestion, doorway, narrow*)	**POIs:** doors, narrow corridors, gates **ROIs:** entrances, hall junctions **Bs:** crowding, yielding, retreating, pushing
Normal Operation (*routine, work, dinner*)	**POIs:** doors, desks, kitchen tables **ROIs:** offices, canteen seating, kitchens **Bs:** move to desk, prepare food, sit, disperse

are relevant for constructing agent behavior trees. The retrieved nodes are injected into the LLM instruction context, together with the semantic representation of the environment derived from `world.yaml`, then combined with the original prompt. The LLM produces a structured intermediate representation specifying agent models (e.g., nurse, businessperson), spawn locations, selected behavior nodes, their parameters, and the execution order of these nodes, all conditioned on the prompt intent. This representation is subsequently compiled into valid behavior trees.

b) Ambient Agent Parsing via LLMs and Diffusion Models: To efficiently generate large populations of *Ambient Agents*, we leverage a pre-trained diffusion model from [45] to generate velocity fields that describe crowd motion. In contrast to the original approach, which relies on CLIP [47] model embeddings and image-based semantic maps to infer start and goal regions, we exploit our structured world representation to obtain semantically precise and scalable group specifications. We observed that the CLIP with semantic map approach does not scale well in environments with many candidate regions, as spatial references such as `top left` or `middle bottom` are weakly grounded and ambiguous. In our pipeline, the processed `world.yaml` specification is injected into the user prompt and provided to the LLM, which extracts structured parameters for each human group, including the center position, width, and length of the start and goal regions, agent models, group sizes, and canonicalized group descriptions [45]. This produces semantically coherent and spatially precise start and goal distributions that are consistent with both the user intent and the environment layout. The canonicalized group descriptions, semantic map image and start and goal distributions are then passed to the diffusion model to generate velocity fields, which are stored and later accessed by the social force model through the procedure described in Algorithm 1. Pedestrians are sampled from the inferred start distributions and instantiated with behavior trees containing the `FollowVelocityField` node, enabling scalable and socially compliant navigation guided by velocity field priors.

IV. Evaluation

Following the evaluation pipelines of [48] and [49], we quantitatively evaluate the quality of generated worlds using our proposed pipeline against open-source baseline approaches, including HouseGAN++ [50], ProcTHOR [35], SceneCraft [51], HouseDiffusion [52], and Structured3D [53] for indoor residential environments, as well as ProcTHOR for office

TABLE III: Quantitative evaluation on text-image alignment and visual-quality preference (↑ higher is better). **Bold** marks the best performance. Visual quality preference indicates Gemini 3.0 Pro and human ratings (scale 1-10) for our method compared to baselines.

Method	**Text-Image Alignment↑**			**Visual-Quality Evaluations (Gemini 3.0 Pro / Human Evaluation)**			
	CLIP↑	BLIP↑	VQA↑	Object Diversity↑	Fidelity↑	Spatial Realism↑	Overall Impression↑
HouseGAN++	0.127	0.095	0.3102	6.61 / 6.57	7.12 / 7.15	6.78 / 6.77	6.92 / 6.58
ProcTHOR	0.134	0.098	0.3885	7.77 / 7.48	7.07 / 6.96	7.47 / 7.08	7.30 / 7.20
SceneCraft	**0.194**	**0.171**	**0.5682**	8.95 / 8.82	**9.45** / 9.12	9.12 / 9.05	**9.42** / 9.15
HouseDiffusion	0.142	0.111	0.4241	**9.35** / 9.05	8.83 / 8.65	8.36 / 8.10	8.64 / 8.35
Structured3D	0.174	0.140	0.5108	8.84 / 8.79	9.25 / **9.48**	**9.62** / **9.55**	9.38 / 8.92
Ours	0.188	0.156	0.5540	9.15 / **9.28**	9.41 / 9.32	9.42 / 9.30	9.12 / **9.45**

environments. These baselines generate static environments using generative models such as diffusion models and GANs, 3D reconstruction techniques, or procedural generation approaches, whereas our pipeline additionally incorporates the previously presented pre- and post-processing semantic RAG systems. We restrict comparisons to approaches that are open-source and fully reproducible, as several existing methods are not publicly available, are not reproducible, or have only limited code access. To evaluate our scenario generation, we compare our approach against the baselines RESCUE [54], which proposes a 3D adaptive social force model, TextCrowd [33], which employs a diffusion-baed generation of scenarios, and the traditional ROS integration of the traditional social force model Pedsim [55]. Our world generation and scenario generation approaches are evaluated separately against their respective state-of-the-art (SotA) models.

A. World Generation Evaluation

For each of the baseline methods, we generated 20 distinct indoor residential environments. For our method, we generated an additional 20 residential and 10 office environments using diverse text prompts covering various room types, furniture arrangements, and styles. To evaluate the quality of the generated worlds, we employed a combination of calculated metrics and human evaluations. These metrics include CLIP [56], BLIP [57], and VQA [58] scores to assess text-image alignment, as well as qualitative metrics focusing on object diversity (Is the scene rich and non-repetitive?), visual and physical plausibility (does each object look like real instance?), scene-level correctness (Does the scene make sense?), and overall impression (How would you rate the scene in general?). Additionally, we utilize a GPT model (Gemini 3.0 Pro) as a neutral evaluator, scoring the model outputs on the qualitative metrics as well as 12 human participants to have a human baseline for comparison. The results are summarized in Table III.

Overall, the results show that our method achieves consistently competitive performance across both text–image alignment and visual-quality metrics, with top or near-top scores in all evaluated categories. While some baselines outperform our method in individual metrics such as SceneCraft in CLIP and VQA, or Structured3D these methods are either specialized for static reconstruction or optimized for specific visual criteria.

B. Scenario Generation Evaluation

To evaluate the flexibility of the framework, we generated seven distinct and commonly occurring scenarios within a generated hospital. Natural language prompts were mapped to semantic behaviors via our system pipeline. Exemplary prompts together with the corresponding generated environments are illustrated in Fig. 5, while the complete set of scenarios is listed in Table II. Additionally, we further examined whether the framework can generate context-aware behaviors by applying identical prompts to multiple environments, including a warehouse, office, and hospital. Four representative prompts were selected—*coffee break*, *door block*, *queuing*, and *emergency* and instantiated in each world. The results are illustrated in Fig. 6. For quantitative evaluations of the scenario generation, we employed the same metrics as in the world generation evaluation, focusing on object diversity, plausibility, and spatial correctness. We also computed the average score for these metrics. Both Gemini 3.0 Pro and human evaluators rated the generated scenarios based on these criteria. The results are summarized in Table IV. Additionally, we compute a CLIP-V score by uniformly sampling frames from the generated scenario videos and averaging their CLIP similarities with the corresponding scenario descriptions to assess consistency between text prompts and visual outputs across temporal evolution of the scene. We compared our method against state-of-the-art scenario generation approaches including RESCUE [54] for emergency scenarios, TextCrowd [33] for social scenarios, and Pedsim [55] for queuing scenarios.

TABLE IV: Quantitative Evaluations of Scenarios

	Meth.	**Divers.**	**Plaus. (Gemini/Human)**	**Correct.**	**Avg (D,P,C)**	**CLIP**
Emgcy.	RESCUE	7.82/**8.95**	**8.44/9.32**	**8.91/9.54**	**8.39/9.27**	**0.188**
	TextCrowd	6.95/7.82	6.23/7.65	7.12/7.94	6.77/7.80	0.154
	Ours	**7.92**/8.55	7.55/8.82	7.85/8.25	7.77/8.54	0.162
Norm.	Pedsim	8.52/8.15	7.45/8.72	7.82/7.95	7.93/8.27	0.165
	TextCrowd	8.95/8.62	7.92/**9.15**	8.25/8.85	8.37/8.87	0.174
	Ours	**9.53/9.42**	**8.55**/9.05	**9.12/9.51**	**9.07/9.33**	**0.181**
Queue	Pedsim	7.25/8.05	7.05/8.22	**7.65**/7.92	7.32/8.06	0.112
	Ours	**8.42/9.15**	**8.12/9.35**	7.45/**8.82**	**8.00/9.11**	**0.159**

The results of table Table IV indicate that our method outperforms the baselines across most of the evaluated scenarios and metrics. Both Gemini 3.0 Pro and human evaluators consistently rated our generated scenarios within

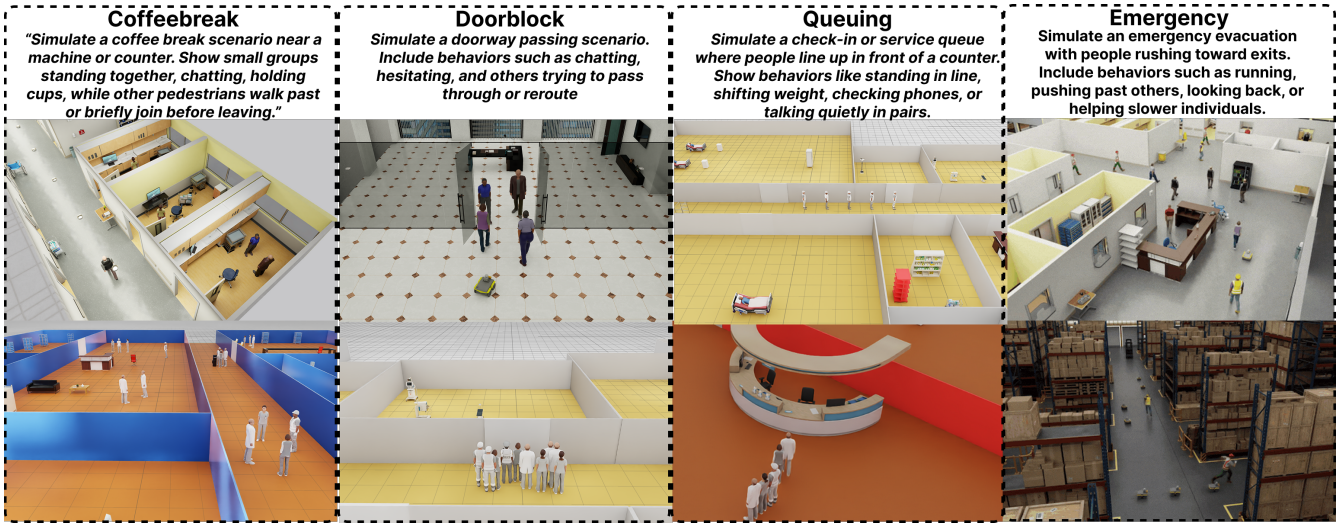

Fig. 6: Exemplary results of four scenario prompts, coffee break, door block, queuing, and emergency, generated across different environments including a warehouse, office, and hospital. Each prompt was entered through the RViz panel, and the resulting behaviors were adapted to the semantic layout of the respective world. For instance, in the hospital environment, queuing behavior emerges at the reception counter, while in a corridor setting the same queuing prompt results in people waiting near a water dispenser. The examples demonstrate how identical prompts can yield diverse and context-aware behaviors depending on the available regions of interest (ROIs), points of interest (POIs), and world geometry.

the normal and queue situations higher in terms of alignment with the prompts, plausibility of pedestrian behaviors, and overall visual quality. The CLIP-V scores further corroborate these findings, demonstrating that our scenarios maintain a stronger and more consistent relationship with the input text prompts throughout their temporal evolution. Only within the rescue scenario, the RESCUE model performed better, which was expected due to it being specifically trained on rescue scenarios.

Furthermore, we illustrate exemplary scenarios generated with our approach in Fig. 6. The results demonstrate that our generated scenarios are not fixed templates but instead adapt to the semantic layout and available objects of each world. For example, the queuing prompt led to different configurations: in the hospital environment, pedestrians formed a line at the reception counter, while in the corridor setting the same prompt resulted in pedestrians waiting near a water dispenser. Similarly, the emergency prompt produced distinct evacuation dynamics: in the compact hospital world with a single exit, all pedestrians rushed toward the same doorway, whereas in the warehouse world with multiple exits, groups split and evacuated in different directions. Coffee break and door block scenarios exhibited comparable variations, with agents gathering near tables or obstructing doorways depending on the spatial affordances.

These findings confirm the success of our pipeline and demonstrate that scenario generated using our system are inherently dependent on semantic regions of interest (ROIs), points of interest (POIs), and world geometry. Rather than producing uniform outcomes, the system leverages world semantics to yield diverse, contextually appropriate, and realistic behaviors from identical prompts. Additional visualizations and animated generations will be provided in the supplementary material.

V. Conclusions

In this work, we presented a ROS 2 framework for generating semantically grounded and dynamic scenarios from text prompts. Unlike existing pipelines that rely on static scenes or simulator-specific models, our approach introduces unified representations of pedestrians and environments, an end-to-end text-to-scenario generation pipeline, and practical interfaces for scalable use. By linking prompts to semantic regions of interest (ROIs) and points of interest (POIs), the system enables behaviors that are both interpretable and reproducible across simulation engines. We evaluated the framework through a set of unseen scenarios against state of the art baselines and achieved competitive results. Furthermore, we conducted a user study with participants from diverse backgrounds. The results demonstrated that the generated environments are realistic, usable, and significantly reduce the effort required to generate fully interactive simulation worlds with humans and crowds. Future work will focus on grounding against real trajectory datasets, tighter coupling with motion-generation models such as *Trace&Pace* [32] or *PedGen* [30].

References

[1] L. Wang, S. W. Kim, J. Yang, C. Yu, B. Ivanovic, S. Waslander, Y. Wang, S. Fidler, M. Pavone, and P. Karkus, "Distillnerf: Perceiving 3d scenes from single-glance images by distilling neural fields and foundation model features," *Advances in Neural Information Processing Systems*, vol. 37, pp. 62 334–62 361, 2024.

[2] Y. Wu, L. Pan, W. Wu, G. Wang, Y. Miao, F. Xu, and H. Wang, "Rl-gsbridge: 3d gaussian splatting based real2sim2real method for robotic manipulation learning," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 192–198.

[3] J. Abou-Chakra, K. Rana, F. Dayoub, and N. Suenderhauf, "Physically embodied gaussian splatting: A visually learnt and physically grounded 3d representation for robotics," in *8th Annual Conference on Robot Learning*, 2024.

[4] M. Wang, Y. Zhang, W. Xu, R. Ma, C. Zou, and D. Morris, "Decoupledgaussian: Object-scene decoupling for physics-based interaction," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 11 361–11 372.

[5] J. Tang, Y. Nie, L. Markhasin, A. Dai, J. Thies, and M. Nießner, "Diffuscene: Denoising diffusion models for generative indoor scene synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 20 507–20 518.

[6] Y. Wang, X. Qiu, J. Liu, Z. Chen, J. Cai, Y. Wang, T.-H. Wang, Z. Xian, and C. Gan, "Architect: Generating vivid and interactive 3d scenes with hierarchical 2d inpainting," *Advances in Neural Information Processing Systems*, vol. 37, pp. 67 575–67 603, 2024.

[7] Y. Yang, B. Jia, P. Zhi, and S. Huang, "Physcene: Physically interactable 3d scene synthesis for embodied ai," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 262–16 272.

[8] M. Deitke, E. VanderBilt, A. Herrasti, L. Weihs, K. Ehsani, J. Salvador, W. Han, E. Kolve, A. Kembhavi, and R. Mottaghi, "Procthor: Large-scale embodied ai using procedural generation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 5982–5994, 2022.

[9] M. Deitke, D. Schwenk, J. Salvador, L. Weihs, O. Michel, E. VanderBilt, L. Schmidt, K. Ehsani, A. Kembhavi, and A. Farhadi, "Objaverse: A universe of annotated 3d objects," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 13 142–13 153.

[10] N. Pérez-Higueras, R. Otero, F. Caballero, and L. Merino, "Hunavsim: A ros 2 human navigation simulator for benchmarking human-aware robot navigation," *IEEE robotics and automation letters*, vol. 8, no. 11, pp. 7130–7137, 2023.

[11] J. Weisz, Y. Huang, F. Lier, S. Sethumadhavan, and P. Allen, "Robobench: Towards sustainable robotics system benchmarking," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 3383–3389.

[12] A. Biswas, A. Wang, G. Silvera, A. Steinfeld, and H. Admoni, "SocNavBench: A grounded simulation testing framework for evaluating social navigation," *ACM Transactions on Human-Robot Interaction*, vol. 11, no. 3, 2022.

[13] N. Tsoi, A. Xiang, P. Yu, S. S. Sohn, G. Schwartz, S. Ramesh, M. Hussein, A. W. Gupta, M. Kapadia, and M. Vázquez, "Sean 2.0: Formalizing and generating social situations for robot navigation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 047–11 054, 2022.

[14] V. Shcherbyna, L. Kästner, D. Diaz, H. G. Nguyen, M. H.-K. Schreff, T. Lenz, J. Kreutz, A. Martban, H. Zeng, and H. Soh, "Arena 4.0: A comprehensive ros2 development and benchmarking platform for human-centric navigation using generative-model-based environment generation," *2025 IEEE International Conference on Robotics and Automation - ICRA 2025*, 2025.

[15] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Physical Review E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[16] J. van den Berg, S. J. Guy, M. C. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1928–1935.

[17] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.

[18] A. Aroor, S. L. Epstein, and R. Korpan, "Mengeros: A crowd simulation tool for autonomous robot navigation." in *AAAI Fall Symposia*, 2017, pp. 123–125.

[19] N. Pérez-Higueras, R. Otero, F. Caballero, and L. Merino, "Hunavsim: A ros 2 human navigation simulator for benchmarking human-aware robot navigation," *IEEE Robotics and Automation Letters*, 2023.

[20] M. Escudero-Jiménez, N. Pérez-Higueras, A. Martínez-Silva, F. Caballero, and L. Merino, "Hunavsim 2.0: An enhanced human navigation simulator for human-aware robot navigation," *arXiv preprint arXiv:2507.17317*, 2025.

[21] S. Xu, Z. Li, Y.-X. Wang, and L.-Y. Gui, "Interdiff: Generating 3d human-object interactions with physics-informed diffusion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 14 928–14 940.

[22] C. Diller and A. Dai, "Cg-hoi: Contact-guided 3d human-object interaction generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 888–19 901.

[23] S. Xu, Y.-X. Wang, L. Gui *et al.*, "Interdreamer: Zero-shot text to 3d dynamic human-object interaction," *Advances in Neural Information Processing Systems*, vol. 37, pp. 52 858–52 890, 2024.

[24] S. Xu, D. Li, Y. Zhang, X. Xu, Q. Long, Z. Wang, Y. Lu, S. Dong, H. Jiang, A. Gupta *et al.*, "Interact: Advancing large-scale versatile 3d human-object interaction generation," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 7048–7060.

[25] M. A. Graule and V. Isler, "Gg-llm: Geometrically grounding large language models for zero-shot human activity forecasting in human-aware task planning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 568–574.

[26] J. Sun, Q. Zhang, Y. Duan, X. Jiang, C. Cheng, and R. Xu, "Prompt, plan, perform: Llm-based humanoid control via quantized imitation learning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 16 236–16 242.

[27] Y. Zhao, Q. Wu, Y. Wang, Y.-W. Tai, and C.-K. Tang, "Navigating motion agents in dynamic and cluttered environments through llm reasoning," *arXiv preprint arXiv:2503.07323*, 2025.

[28] X. Linghu, J. Huang, X. Niu, X. S. Ma, B. Jia, and S. Huang, "Multi-modal situated reasoning in 3d scenes," *Advances in Neural Information Processing Systems*, vol. 37, pp. 140 903–140 936, 2024.

[29] W. Zu, W. Song, R. Chen, Z. Guo, F. Sun, Z. Tian, W. Pan, and J. Wang, "Language and sketching: An llm-driven interactive multimodal multitask robot navigation framework," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 1019–1025.

[30] Z. Liu, J. Lin, W. Wu, and B. Zhou, "Learning to generate diverse pedestrian movements from web videos with noisy labels," in *The Thirteenth International Conference on Learning Representations*, 2024.

[31] W. Wu, H. He, J. He, Y. Wang, C. Duan, Z. Liu, Q. Li, and B. Zhou, "Metaurban: An embodied ai simulation platform for urban micromobility," *International Conference on Learning Representation*, 2025.

[32] D. Rempe, Q. Fu, M. Xu *et al.*, "Trace and pace: Controllable human motion generation via diffusion priors," *arXiv preprint arXiv:2408.11169*, 2024.

[33] X. Ji, Z. Pan, X. Gao, and J. Pan, "Text-guided synthesis of crowd animation," in *ACM SIGGRAPH 2024 Conference Papers*, 2024, pp. 1–11.

[34] A. Panayiotou, P. Charalambous, and I. Karamouzas, "Gen-c: Populating virtual worlds with generative crowds," *arXiv preprint arXiv:2504.01924*, 2025.

[35] M. Deitke *et al.*, "ProcTHOR: Large-scale embodied ai using procedural generation," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[36] C. Wang *et al.*, "Holodeck: Language guided generation of 3d embodied ai environments," *arXiv preprint arXiv:2312.09067*, 2023.

[37] International Code Council, "International residential code," https://codes.iccsafe.org/content/IRC2021P3, 2021, section R304.1: Minimum Room Areas.

[38] UK Department for Communities and Local Government, "Technical housing standards – nationally described space standard," https://www.gov.uk/government/publications/technical-housing-standards-nationally-described-space-standard, UK Government, Tech. Rep., March 2015, minimum gross internal floor areas and storage.

[39] British Council for Offices, "Bco guide to specification," British Council for Offices, London, UK, Tech. Rep., 2019, workplace density standards and office space planning guidelines.

[40] NHS England, "Health building note 03-01: Adult inpatient facilities," https://www.england.nhs.uk/publication/health-building-note-03-01-adult-in-patient-facilities/, Department of Health and Social Care, Tech. Rep., 2021, single bedroom minimum area requirements for healthcare facilities.

[41] Australasian Health Infrastructure Alliance, "Australasian health facility guidelines: Part b – health facility briefing and planning," https://healthfacilityguidelines.com.au/, Australasian Health Infrastructure Alliance, Tech. Rep., 2023, hPU 340 Intensive Care Unit room specifications.

[42] S. Gillies, C. van der Wel, J. Van den Bossche, M. W. Taves, J. Arnott, B. C. Ward *et al.*, "Shapely (2.1.2)," 2025. [Online]. Available: https://doi.org/10.5281/zenodo.17193310

[43] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz, "The walking behaviour of pedestrian social groups and its impact on crowd dynamics," *PLOS ONE*, vol. 5, no. 4, pp. 1–7, 04 2010. [Online]. Available: https://doi.org/10.1371/journal.pone.0010047

[44] M. Moussaïd, D. Helbing, S. Garnier, A. Johansson, M. Combe, and G. Theraulaz, "Experimental study of the behavioural mechanisms underlying self-organization in human crowds," *Proceedings of the Royal Society B: Biological Sciences*, vol. 276, no. 1668, pp. 2755–2762, 2009.

[45] C. Ji, Y. Zhang, G. Yi, and collaborators, "Text-guided synthesis of crowd animation," *ACM Transactions on Graphics (SIGGRAPH Asia)*, 2024.

[46] J. Van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *2008 IEEE International Conference on Robotics and Automation*. IEEE, 2008, pp. 1928–1935.

[47] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[48] H. I. D. Pun, H. I. I. Tam, A. T. Wang, X. Huo, A. X. Chang, and M. Savva, "Hsm: Hierarchical scene motifs for multi-scale indoor scene generation," *arXiv preprint arXiv:2503.16848*, 2025.

[49] H. I. I. Tam, H. I. D. Pun, A. T. Wang, A. X. Chang, and M. Savva, "SceneEval: Evaluating semantic coherence in text-conditioned 3D indoor scene synthesis," 2025.

[50] N. Nauata, S. Hosseini, K.-H. Chang, H. Chu, C.-Y. Cheng, and Y. Furukawa, "House-gan++: Generative adversarial layout refinement network towards intelligent computational agent for professional architects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 632–13 641.

[51] X. Yang, Y. Man, J. Chen, and Y.-X. Wang, "Scenecraft: Layout-guided 3d scene generation," *Advances in Neural Information Processing Systems*, vol. 37, pp. 82 060–82 084, 2024.

[52] M. A. Shabani, S. Hosseini, and Y. Furukawa, "Housediffusion: Vector floorplan generation via a diffusion model with discrete and continuous denoising," *arXiv preprint arXiv:2211.13287*, 2022.

[53] J. Zheng, J. Zhang, J. Li, R. Tang, S. Gao, and Z. Zhou, "Structured3d: A large photo-realistic dataset for structured 3d modeling," in *European Conference on Computer Vision*. Springer, 2020, pp. 519–535.

[54] X. Liu, T. Zhou, H. Kang, J. Ma, Z. Wang, J. Huang, W. Weng, Y.-K. Lai, and K. Li, "Rescue: Crowd evacuation simulation via controlling sdm-united characters," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 24 955–24 964.

[55] "pedsim_ros," https://github.com/srl-freiburg/pedsim_ros, accessed: 2026-01-22.

[56] J. Hessel, A. Holtzman, M. Forbes, R. Le Bras, and Y. Choi, "Clipscore: A reference-free evaluation metric for image captioning," in *Proceedings of the 2021 conference on empirical methods in natural language processing*, 2021, pp. 7514–7528.

[57] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*. PMLR, 2023, pp. 19 730–19 742.

[58] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2425–2433.