TANGO: Traversability-Aware Navigation with Local Metric Control for Topological Goals

Stefan Podgorski*[1], Sourav Garg*[1], Mehdi Hosseinzadeh[1], Lachlan Mares[1], Feras Dayoub[1], Ian Reid[1,2]

***Abstract*— Visual navigation in robotics traditionally relies on globally-consistent 3D maps or learned controllers, which can be computationally expensive and difficult to generalize across diverse environments. In this work, we present a novel RGB-only, object-level topometric navigation pipeline that enables zero-shot, long-horizon robot navigation without requiring 3D maps or pre-trained controllers. Our approach integrates global topological path planning with local metric trajectory control, allowing the robot to navigate towards object-level sub-goals while avoiding obstacles. We address key limitations of previous methods by continuously predicting local trajectory using monocular depth and traversability estimation, and incorporating an auto-switching mechanism that falls back to a baseline controller when necessary. The system operates using foundational models, ensuring open-set applicability without the need for domain-specific fine-tuning. We demonstrate the effectiveness of our method in both simulated environments and real-world tests, highlighting its robustness and deployability. Our approach outperforms existing state-of-the-art methods, offering a more adaptable and effective solution for visual navigation in open-set environments. The source code is made publicly available: `https://github.com/podgorki/TANGO`.**

I. INTRODUCTION

Visual navigation is a fundamental challenge in robotics, with significant implications for autonomous agents operating in real-world environments. Traditional approaches often rely on constructing precise, globally consistent geometric 3D maps [1]–[3], which can be computationally intensive and difficult to generalize across diverse settings. Alternatively, methods designed for navigating in previously unseen environments [4], [5] may not effectively leverage prior knowledge, limiting their efficiency and adaptability.

Inspired by human navigation abilities – where we can traverse environments by reasoning over previously observed images or objects without detailed 3D maps – visual topological navigation has emerged as a promising alternative [6]–[8]. Recent research has predominantly focused on *image-level* topological maps [6], [7], which, while straightforward, have limited representational capacity. They often lack semantic richness and are sensitive to viewpoint changes, hindering their applicability in dynamic and diverse environments.

In contrast, *object-level* topological maps [8] offer several advantages, including direct open-set natural language querying, semantic interpretability [9], and viewpoint-invariant visual recognition [10]. These attributes are crucial for enabling open-world navigation that can be seamlessly deployed across different environments, tasks, and robotic platforms. However, integrating object-level topological information into navigation pipelines presents challenges, particularly in bridging global planning with local motion control while ensuring obstacle avoidance and traversability.

Fig. 1. We present a topometric navigation pipeline that uniquely bridges *topological* global path planner and *metric* local trajectory planning, without needing 3D maps or learnt controllers. This enables our method to effective avoid obstacles (bottom row) even when no such objects were present in the mapping (teach) run.

In this work, we present a novel RGB-only, object-level, *topometric* navigation pipeline for zero-shot robot control, in contrast with recent learnt controllers [6], [7], [11]. Specifically, we propose a unique integration of global path planning and local motion planning, where a robot *metrically* plans its motion to move towards *topologically* planned object-level sub-goals. The latter is achieved through a recent work, RoboHop [8], where its global path planner generates object-level sub-goal cost mask for robot's current observation (see Figure 2). While this sub-goal mask can guide a robot for *where to head*, it does not account for traversability or obstacle avoidance due to its purely topological nature. We address this limitation through our proposed topometric controller, where we explicitly predict traversable image segments, project them in Bird's Eye View (BEV) space using monocular metric depth, plan a trajectory to the farthest least-cost sub-goal, and continue this process until the long-horizon goal is reached.

The contributions of this paper are as follows:

- a novel topometric controller that uniquely bridges *topological* global path planner and *metric* local trajectory planning to enable long-horizon object-goal navigation without needing 3D maps or learnt controllers;
- an RGB-only method to *continuously* predict *local trajectory* using single-view depth and traversability;
- an auto-switch-control approach that switches from our proposed controller to a fallback controller by detecting

[1]Australian Institute for Machine Learning (AIML), The University of Adelaide, Australia.
[2] Mohamed Bin Zayed University of Artificial Intelligence, UAE.
*Equal contirbution.

the absence of visible traversable regions; and
- a real-world demonstration (5 Hz) of our modular navigation pipeline built on top of 'foundation models' such as Fast Segment Anything [12], [13], Depth Anything [14], and CLIP [15], which *explicitly* maintains an open-set applicability.

II. RELATED WORKS

A. Topological sub-goals for Navigation

A vast majority of vision-based navigation methods rely on 3D maps [2]–[4], [16]–[18], where significant progress has been made both for 'unseen' [19]–[22] and prior map-based 'seen' [16], [23], [24] environments. In contrast, a range of methods exist that directly use visual topological sub-goals for long-horizon navigation, without requiring a 3D map. Inspired by human-like navigation capability, SPTM [6] demonstrated a learning-based navigation controller using an image sequence as a map. Recent works in this direction have innovated in real-world training and deployment [25], use of language [26], adaptation to different embodiments [7], and jointly learning to explore [27]. These methods predominantly use an image as a sub-goal, which has to have been captured from a different robot pose to obtain a control signal from the image pair. This can either be achieved through learning-based approaches [6], [16], [26], [28]–[30] or visual servoing [31]–[39]. Instead of using image sub-goals, recent methods such as PixNav [11] and RoboHop [8] proposed to use respectively a pixel and an object as a sub-goal – visible in the robot's current observation. While PixNav learns a controller in simulation for this purpose, RoboHop uses a zero-shot 'segment servoing' approach to reach object sub-goals. In this work, we follow RoboHop's open-set navigation pipeline to obtain object sub-goals using its global path planner based on object-level topological graph, and propose to combine this with a traversability-aware trajectory planner to achieve a more performant navigation system.

B. Open-set, Zero-shot, Large Models-enhanced Navigation

There has been significant progress in learning-based navigation using both reinforcement [40]–[42] and imitation [43], [44], across different application areas including autonomous driving in structured environments [45], [46], off-road outdoor settings [25], [47], and aerial vehicles [23], [48]–[50]. With recent advances in large-scale general (foundation) models for perception, researchers have now focused on leveraging such models for their open-set characteristics and zero-shot applicability. Examples include ZSON [51], COW [52], GOAT [53], [54], and VL-Maps [16], which rely on joint vision-language embedding space of CLIP [15] for open-vocabulary navigation. Although enabling open-set goal description in natural language, most of these methods rely on learning-based techniques for robot control. In the same vein of using foundation models, Large Language Models (LLMs) have been explored for zero- or few-shot navigation, e.g., NavGPT [55], [56], MapGPT [57], and VisionGPT [58]. More recently, multimodal LLMs have also been leveraged for navigation using videos, e.g., NaVid [59] and MobilityVLA [60], and visual annotations, e.g., PIVOT [61] and CoNVOI [62]. While these methods aim to directly control the robot actions, they are limited in terms of their grounding in the robot's map [55], 3D spatial understanding [61], and long inference times [59], [60]. Distinct from the aforementioned techniques, we build a *modular* open-set navigation pipeline to generate zero-shot control signal by using Segment Anything Model (SAM) [13] for object-level topological mapping and planning; CLIP [15] combined with SAM for traversability estimation; and Depth Anything [14] for monocular depth estimation of traversable segments for local motion planning in BEV space.

C. Teach & Repeat and Experiential Navigation

A significant subset of navigation literature deals with visual teach-and-repeat task [63]–[71]. These methods typically do not require 3D maps for navigation, as they implicitly leverage the inherent assumption of the 'narrower' task of repeating the teach run by simply using image-based visual servoing. A more generalized version of such navigation can be referred to as experiential learning of robot navigation [72]. By learning from real-world data at large-scale, e.g., ViNT – a foundation model for navigation [73], such control policies can exhibit general understanding of traversability, reachability, and exploration objectives. Although more capable than teach-and-repeat, the end-to-end learning paradigm of such methods limits their interpretability, and induces control-related data bias that can limit their broader applicability. In contrast to learning 'foundational control', we aim to leverage 'vision foundations for navigation', which presents several benefits: open-set, object-level queryable map; interpretable object-level global plan; and explicit traversability-aware local motion planning. This also enables a novel capability: reaching *seen but unvisited* object goals, which steps beyond simple teach-and-repeat task while not requiring any learning.

III. APPROACH

Our proposed method aims to effectively integrate the robot's understanding of topologically-connected object sub-goals with an ability to reach that sub-goal through traversability-aware instantaneous trajectory planning. As the topological global planner continuously updates the sub-goal masks, the metric local planner enables movement through the traversable paths to continuously reach the updated sub-goals until the final goal is reached. We present our topometric controller in the following subsections, where we first provide background details for the topological mapping, localization and planning pipeline based on RoboHop [8], and then describe the novel integration of our local metric motion planner.

A. Topological Object-based Mapping and Planning

We define the topological map of the environment as a graph $\mathscr{G} = (\mathscr{N}, \mathscr{E})$, where $\mathscr{N}$ and $\mathscr{E}$ denote the set of nodes and edges, respectively. Each node n_i in $\mathscr{G}$ corresponds to an image segment $\mathbf{M}_i$, which represents a meaningful object.

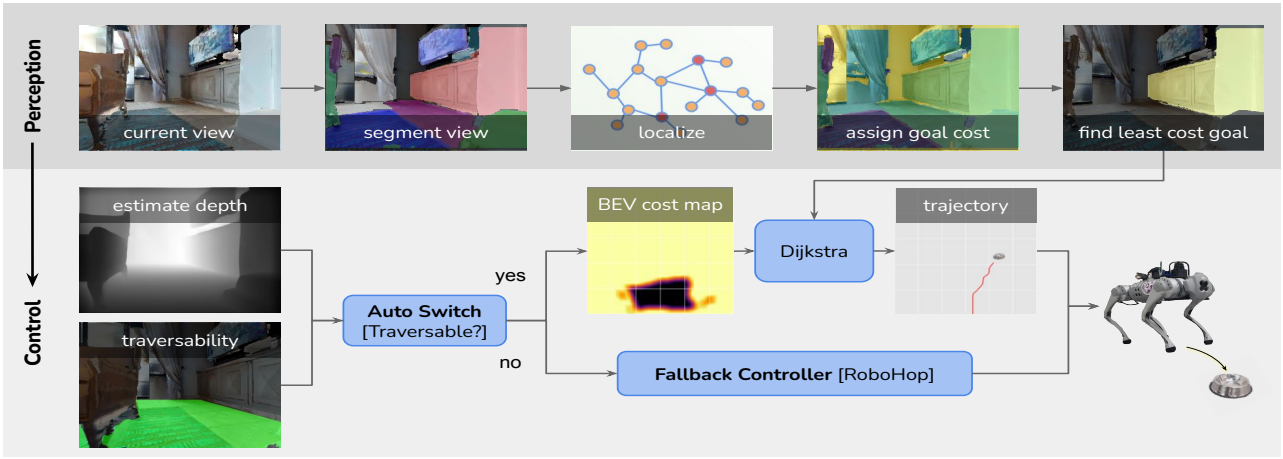

Fig. 2. TANGO's Navigation Pipeline. **Perception**: The robot's current view is segmented using a foundational segmentation model (SAM), the segments are localised within an object-level topological map using local feature matching (LightGlue). Each segment is assigned a cost based on its topological proximity to the final goal segment, the segment closest to the final goal is selected to drive the controller. **Control**: A BEV traversability map is computed by combining state-of-the-art depth estimation with open-set text query capabilities (CLIP) to identify traversable surfaces such as 'floor' or 'ground'. This depth and semantic information is integrated to generate a BEV cost map (yellow high cost, black low cost). Dijkstra's algorithm is applied to compute the shortest path to the sub-goal segment, providing a trajectory that avoids obstacles and generates yaw control signals for robot navigation. This perception-action loop is repeated continuously until the robot reaches the final goal object.

An edge $e_{ij} \in \mathscr{E}$ connects image segments and is defined as either: a) *intra-image edges*, which connect centroids $\mathbf{M}_i$ and $\mathbf{M}_j$ within the same image I^t using Delaunay Triangulation, or b) *inter-image edges* which connect corresponding segments matched across different images through data association.

Mapping: The segmentation masks are extracted from an image sequence $\{I^t\}$ using a foundational model, such as SAM [13]. The zero-shot capabilities of these recent foundational models are particularly valuable, as they enable us to construct a topological representation that is not restricted to a closed-world assumption of predefined objects. Moreover, these models inherently support integration with richer descriptors and language models, allowing for more expressive scene understanding. For node/segment tracking during the mapping process, we utilise local feature matching, which was observed to perform better than DINOv2-based matching in the original RoboHop [8]. Specifically, we extract SuperPoint [74] features and match them using LightGlue [75] to identify pixel-level correspondences between an image pair. These matches are converted to segment-level correspondences based on the membership of the pixels in their respective segmentation masks.

Localisation: At every step, the robot localises itself within a temporal window of map images centered around its previous estimation of localised reference image index. Given the candidate map images, the robot's current image is matched pairwise using the same local feature matching process described in the mapping section above. This provides segment-level correspondences between the current image and the map images. Using these correspondences, we obtain sub-goal costs for each of the query segments using the global planner, as described next.

Global Planning: By leveraging the connectivity between segments in the map, we compute path lengths between the localized reference map segments and the goal segment. To facilitate this, we assign edge weights between the source and destination nodes: specifically, inter-image edges (connecting segments from different images) are assigned a weight of 0 (being the same object instance), while intra-image edges (connecting segments within the same image) are given a weight of 1. We then compute a weighted shortest path to the target node in the map using Dijkstra's algorithm, starting from every localized query segment. This yields a sub-goal cost mask for the robot's current observation (see Figure 2), highlighting the desirable objects that the robot should approach to reach its long-horizon goal.

B. Metric Control To Reach Object Sub-Goals

Given the object-level sub-goals planned topologically by the global planner, TANGO generates a local metric motion plan to navigate toward these sub-goals. The transition from topological sub-goals to metric sub-goals is accomplished by computing a BEV traversability map. Using state-of-the-art models, our method combines single-view depth estimation with open-set text query capabilities, enabling the refinement of traversable regions based on object semantics.

Metric BEV Traversability: At each timestep the robot's RGB image is converted to binary segment masks using a foundational model such as SAM [13]. Each segment is assessed for traversability by utilising CLIP [76] text queries, filtering out segments based on their "semantics", such as floor, ground, or rug. Segments within the segment map are set to 1 if assessed as traversable and 0 if non-traversable forming a binary traversability mask. This open-set queryable filter enables fine-grained selection of traversable regions, adaptable to different real-world scenarios. For each sub-goal node segment n_i, we select the lowest-cost image segment $\mathbf{M}_i$ as the representative sub-goal. Once the traversable segment masks and the sub-goal segment are identified, we apply

monocular depth estimation via Depth-Anything [14] to project the traversable segments and sub-goal points into 3D space, resulting in the final metric BEV traversability map. The final sub-goal-point is then selected as the farthest projected point contained in the sub-goal-segment.

Trajectory and Motion Planning: For each input RGB image frame, the metric BEV traversability map is calculated and converted to a cost map for planning. The cost map is formed by applying a distance transform from the traversability masks edges, which is then smoothed with a box filter. Within this cost map, the shortest path to the local 3D sub-goal is determined using Dijkstra's algorithm, generating a series of traversable waypoints along the trajectory to the sub-goal. These waypoints are then used to generate control signals, controlling the robot's yaw angle with a proportional line following controller and holding the linear velocity fixed to effectively navigate toward the sub-goal.

Auto Switch Control: In situations where metric traversability prediction is unreliable or unavailable, such as when the robot is too close to a wall or obstructed by an obstacle, the local controller automatically switches to RoboHop's [8] fully topological "segment servoing" approach. In the absence of reliable traversable regions, this method converts the horizontal pixel offset of each sub-goal mask into yaw velocity (θ) using Equation 1, ensuring the robot can still navigate effectively in these challenging scenarios.

$$\theta = \frac{G}{W}\sum_i w_i(u_i - c_i); \quad w_i = \frac{e^{\tau l_i}}{\sum_i e^{\tau l_i}} \tag{1}$$

where c_i represents the image centre, u_i is the centroid of the segment $\mathbf{M}_i$, l_i is the path length (min-max normalized across localized query segments), τ is the temperature parameter (set to 5), w_i is the softmax weight per query segment, W is the image width, and G is the controller gain (set to 0.4).

IV. EXPERIMENTS

A. Dataset and Evaluation

We use Habitat-Matterport 3D Dataset (HM3D) [77] to evaluate our proposed method. Specifically, we use the validation set of the InstanceImageNav (IIN) challenge [78] that comprises 36 unique environments. We sample 3 episodes (with unique object goals) per environment to benchmark across 108 episodes. For each episode, we use the simulator's path finding method to obtain a mapping (teach) traverse, which is used to construct the object-level topological graph and is available during inference to all the methods for generating sub-goal costs.

We evaluate a controller's ability to navigate to an object goal in a given episode. We report the average success rate, where an episode is deemed successful if the robot is within $1m$ [79] of the goal position, taking maximum 500 steps. The evaluation is repeated based on the starting point of the robot by varying the geodesic length of trajectories. While PixNav [11] only uses two short variations of episode lengths, we further include the full length of the episodes as a more challenging setting. We refer to these as 'easy', 'hard' and 'full', with their starting distance from the goal respectively as 1-3m, 3-5m, and 8-10m. We provide all implementation details in the supplementary video, along with real-world demonstrations.

B. Baselines

We use the following baselines to assess the effectiveness of our proposed method.

1) Ground Truth Goal Masks: We consider two key variants of our navigation pipeline where we use ground truth information for perception and planning to generate goal masks corresponding to robot's image observation. i) `GT-Metric`: we use simulator's semantic instance masks, depth and navigation mesh to find shortest (geodesic) paths from each object instance to the goal object. This provides an accurate metric estimate of object sub-goal costs in robot's current view, thus being the ideal goal mask input for the controller. ii) `GT-Topological`: we use simulator's semantic instance masks to create an object-level topological map (as described in Section III-A), which assumes segmentation, matching/association, and localization to be solved. This object-level map is then used to compute global path lengths, thus the goal masks so obtained lack geometric understanding of object segments layout and only rely on intra-image and inter-image object connectivity. This setting aids in testing the role of planning as well as control while assuming perception to be solved.

2) Robohop: We use the original RoboHop controller as described in Eq. 1, where the goal mask information is used in the form of pixel centers of the object segments weighted by their path lengths.

3) Pixel-guided Navigation (PixNav): PixNav is a transformer-based imitation learning local navigation method [11] that uses a patch of goal pixels that correspond to the final or intermediate navigation goal points. The goal patch is initially fed into the model as a mask with the corresponding RGB image and then executes an action from the discrete action space: Stop, MoveAhead, TurnLeft, TurnRight, LookUp, LookDown. At each subsequent step the current RGB image and a collision signal are used with the history of the previous images and the initial goal mask to predict the next action.

PixNav is a discrete controller with a move-able camera whereas RoboHop and TANGO are continuous with a fixed camera. Accounting for these differences and noting the intended design of PixNav, evaluations for the PixNav controller were set to initialize a viewable intermediate goal given by the topological global planner, where the goal was updated when the method outputted 'Done' or when its memory buffer was full.

V. RESULTS

A. Benchmark comparison

Table I presents a comparison of our proposed method against baseline techniques for varying lengths of trajectories. We also include ablative results where we swap the perception and planning modules with their ground truth counterparts. All compared methods are provided the

TABLE I

NAVIGATION SUCCESS RATE ACROSS VARYING TRAJECTORY LENGTHS.

Controller	Easy [1-3m]	Hard [3-5m]	Full [8-10m]
	GT-Metric		
RoboHop [8]	93.14	78.43	42.16
PixNav [11]	65.69	44.12	15.69
TANGO (ours)	**94.12**	**90.20**	**48.04**
	GT-Topological		
RoboHop [8]	**78.43**	58.82	25.49
PixNav [11]	60.78	44.12	15.69
TANGO (ours)	74.51	**65.69**	**30.39**
	No-GT		
RoboHop [8]	43.56	34.56	13.73
PixNav [11]	51.96	39.22	14.0
TANGO (ours)	**61.76**	**43.14**	**21.57**

TABLE II

ABLATION NAVIGATION SUCCESS RATE FOR THE TANGO CONTROLLER ACROSS 'HARD' 3-5M TRAJECTORIES.

Perception	Control	Success Rate
Segment + Matcher	Depth + Trav	
Sim	Sim	65.69
FastSam + LGlue	Sim	47.95
FastSam + LGlue	DepthAnything + FastSam	43.14

TABLE III

AUTO SWITCH CONTROL IMPROVES TANGO'S SUCCESS RATE.

Control Type	No Switch	Auto Switch	Improvement
Hard [3-5m]	62.14	73.78	**11.64**

same goal information to observe the control performance differences in isolation. For GT-Metric and GT-Topological settings, TANGO uses simulator's depth and traversability estimation, we ablate this in Section V-B.1 for further insights. It can be observed that our proposed method TANGO outperforms both the baselines – learning-based controller PixNav and traversability-unaware zero-shot controller RoboHop – by a significant margin in most cases. The trend remains consistent across easy, hard, and full length trajectories. Furthermore, as the extent of ground truth information reduces from `GT-Metric` through `GT-Topological` to `No-GT`, a general trend of gradual performance decline is observed. For the `GT-Metric` setting, 'perfect' sub-goal masks lead to the highest navigation success rate for all methods, as expected. For the `GT-Topological` setting, performance drops become large for the hard and full-length trajectories. These results highlight the impact of topologically computed global path lengths in contrast with `GT-Metric` which assumes access to a full geometric 3D map/simulator. In the `No-GT` setting, although the absolute performance of all methods is low, the comparison to its `GT` counterparts shows that the performance drop is attributed more to perception than topological planning and control.

B. Ablation Studies

1) Control with and without GT: In Table II, we compare different versions of our proposed controller by ablating ground truth (simulator) components of perception (segmentation and association) and control (depth estimation and traversability) with their respective prediction methods. It can be observed that our full pipeline (last row) only suffers 5% performance drop in comparison to the simulated control components (depth and traversability), whereas perception (and localization) prediction leads to a 18% drop (first to second row). These comparisons emphasize the need for improved segmentation and matching methods more than monocular depth estimation for the downstream task of navigation.

2) Auto Switch Control: The proposed local controller creates a relative bird's-eye-view based on the traversability of the scene at each step. This enables traversal around objects blocking robot's path. However, in tight spaces of a house, there exists situations where the controller is unable to perceive traversable segments, or the traversability estimation fails. In these cases, object segments can still be observed and matched to obtain a valid goal to control the robot's yaw. Thus, the proposed local motion planner switches to the RoboHop controller as fallback, which rotates the robot towards the goal until the traversable segments are visible again for the proposed controller to take over. Table III presents a comparison of our local motion planner with and without the proposed auto switching. The controller variations were evaluated in the `GT-metric` 'hard' setting for a maximum of 250 simulation steps, differing from the other evaluations which use 500 steps. It can be seen that in times of unknown traversability, having the ability to fall back to "segment servoing" enables continual progress towards the goal, thus improving the success rate.

C. Reaching 'seen but unvisited' Goals

Our proposed method uses a *topological* prior map based on a single trajectory. The objects on the way can thus be assumed to be reachable. However, several more objects (in other rooms/places) are observed throughout the mapping

TABLE IV

REACHING SEEN-BUT-UNVISITED GOALS.

Goals Type	Hard [3-5m]	Full [8-10m]
Teach Goals	43.14	21.57
Alt Goals	50.54	25.84

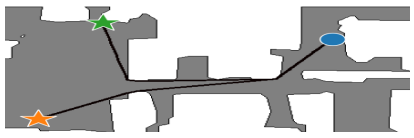

Fig. 3. Seen-but-Unvisited Goals: An example episode with agent's starting position marked in blue, original goal in green and the new seen-but-unvisited goal in orange. The robot has no 'prior experience' of reaching the new (orange) goal, as the map run is traversed from blue to green.

Fig. 4. Successful sequence showing the controller navigating around the couch and between the table successfully arriving at the final goal chair (highlighted in green box)

Fig. 5. Unsuccessful sequence showing the controller navigating around into the bathroom and incorrectly turning towards the vase at step 65 rather than correctly steering towards the toilet - goal location is indicated with green arrow. Lower right side figure shows overlaid (red) goal segment which steers the controller to the left rather than the right.

run, which the robot may not have 'prior experience' of reaching. These *seen-but-unvisited* objects can be selected as a new alternate goal – reaching these goals using only an image-level connectivity may not be possible, whereas object-level connectivity enables identifying and reaching these goals through a traversability-aware local metric controller. Thus, stepping beyond the typical teach-and-repeat paradigm, we evaluate our method's navigation success in reaching *seen-but-unvisited* long-horizon object goals. For each of the episodes, we obtain such goals, referred to as `Alt Goals`, through a simple measure: an object instance (excluding the `wall` and `ceiling` class) is chosen from the last 30% the episode's map (teach) run poses such that the sum of object's average depth from a given pose and its geodesic distance to the original goal is the highest across all possible instances. Thus, the `Alt Goals` get sampled from different rooms or in the same room but far-off from the original goal, as shown in Figure 3. We evaluate on this new task exactly as described for the vanilla task except that the original object goals are replaced by these new goals.

In Table IV, it can be observed that TANGO's success rate for reaching seen-but-unvisited `Alt Goals` is comparable to that for `Teach Goals`, across both 'hard' and 'full' setting. While this clearly demonstrates the capability of our proposed pipeline beyond simple teach-and-repeat, the performance comparison highlights that low performance is not attributed to the difference in the task but to the limitations of perception and planning, as established in Table I. Overall, these results emphasize the role of *object-level* (thus, traversability-aware) topological maps for navigation as opposed to their image-level counterparts, where the latter's lack of an explicit reasoning of objects and traversability limits its goal-reaching to only the poses the map images were captured from.

D. Qualitative Analysis

Under successful trajectories such as shown in Figure 4 the controller displays desirable behaviours such as, an ability to turn correctly, avoid objects and choose a path between objects that are close together such as a coffee table and a couch. On occasion misleading goals from the topological global planner will cause the robot to move in a less optimal direction which results in the robot moving around for an increased number of steps. However, a common failure mode of the local planner is shown in Figure 5. In this particular instance, the controller successfully traverses towards the goal from one room turning into another, following through this room until reaching the final bathroom. In the bathroom, the controller continues to plan and control correctly until an erroneous goal segment influences the controller to turn left rather than right shown in the lower right in white. This error mode highlights the importance of high quality matches between the current view segments and the topological map.

VI. CONCLUSIONS

Topological visual goals based navigation using a single RGB camera is an appealing alternative to classical methods based on 6-DoF pose estimation and geometrically-precise 3D maps. This paper presents a novel topometric navigation controller that bridges object-level topological global planning with traversability-aware local metric motion planning using instantaneous monocular depth. This unique integration built on top of 'vision foundation models' leads to a more readily-deployable navigation system, which performs significantly better than the previous methods including both a learnt and zero-shot controller. Consequently, we demonstrate an interesting a new navigation capability of reaching 'seen-but-unvisited' object goals, which emphasizes the importance of a ground-up object-level navigation pipeline. Furthermore, we demonstrate real-world experiments showcasing obstacle avoidance under significant changes in the map (teach) run.

There are a few limitations of our pipeline which lead to navigation failures: a) Perception: incorrect matching of segments from the current view to the reference segment map leads to incorrect sub-goals; b) Planning: pure-topology based edges in the map graph lack the ability to geometrically disambiguate the relevance of different sub-goals in the current image; and c) Traversability: text- and segmentation-based estimation, although convenient, is prone to errors which leads to the use of a fallback controller. However, as opposed to end-to-end learnt controllers, the modular nature of our proposed pipeline allows drop-in replacement of different components as more performant perception models become rapidly available in future.

REFERENCES

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[2] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. Kelly, and A. J. Davison, "Slam++: Simultaneous localisation and mapping at the level of objects," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 1352–1359.

[3] Q. Gu, A. Kuwajerwala, S. Morin, K. Jatavallabhula, B. Sen, A. Agarwal, C. Rivera, W. Paul, K. Ellis, R. Chellappa, C. Gan, C. de Melo, J. Tenenbaum, A. Torralba, F. Shkurti, and L. Paull, "Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning," in *arXiv*, 2023.

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

[5] Y. Hong, Y. Zhou, R. Zhang, F. Dernoncourt, T. Bui, S. Gould, and H. Tan, "Learning navigational visual representations with semantic map supervision," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3055–3067.

[6] N. Savinov, A. Dosovitskiy, and V. Koltun, "Semi-parametric topological memory for navigation," *arXiv preprint arXiv:1803.00653*, 2018.

[7] D. Shah, A. Sridhar, A. Bhorkar, N. Hirose, and S. Levine, "Gnm: A general navigation model to drive any robot," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 7226–7233.

[8] S. Garg, K. Rana, M. Hosseinzadeh, L. Mares, N. Suenderhauf, F. Dayoub, and I. Reid, "Robohop: Segment-based topological map representation for open-world visual navigation," in *2024 International Conference on Robotics and Automation (ICRA)*. IEEE, 2024.

[9] N. V. Keetha, M. Milford, and S. Garg, "A hierarchical dual model of environment-and place-specific utility for visual place recognition," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6969–6976, 2021.

[10] K. Garg, S. S. Puligilla, S. Kolathaya, M. Krishna, and S. Garg, "Revisit anything: Visual place recognition via image segment retrieval," in *European Conference on Computer Vision (ECCV)*, September 2024.

[11] W. Cai, S. Huang, G. Cheng, Y. Long, P. Gao, C. Sun, and H. Dong, "Bridging zero-shot object navigation and foundation models through pixel-guided navigation skill," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 5228–5234.

[12] X. Zhao, W. Ding, Y. An, Y. Du, T. Yu, M. Li, M. Tang, and J. Wang, "Fast segment anything," 2023.

[13] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," 2023.

[14] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *CVPR*, 2024.

[15] 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.

[16] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, London, UK, 2023.

[17] A. Rosinol, A. Violette, M. Abate, N. Hughes, Y. Chang, J. Shi, A. Gupta, and L. Carlone, "Kimera: From slam to spatial perception with 3d dynamic scene graphs," *The International Journal of Robotics Research*, vol. 40, no. 12-14, pp. 1510–1546, 2021.

[18] G. Georgakis, B. Bucher, K. Schmeckpeper, S. Singh, and K. Daniilidis, "Learning to map for active semantic goal navigation," *arXiv preprint arXiv:2106.15648*, 2021.

[19] J. Zhang, L. Dai, F. Meng, Q. Fan, X. Chen, K. Xu, and H. Wang, "3d-aware object goal navigation via simultaneous exploration and identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 6672–6682.

[20] Q. Zhao, L. Zhang, B. He, H. Qiao, and Z. Liu, "Zero-shot object goal visual navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 2025–2031.

[21] V. S. Dorbala, J. F. Mullen Jr, and D. Manocha, "Can an embodied agent find your "cat-shaped mug"? llm-based zero-shot object navigation," *IEEE Robotics and Automation Letters*, 2023.

[22] D. S. Chaplot, D. P. Gandhi, A. Gupta, and R. R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4247–4258, 2020.

[23] S. Weiss, D. Scaramuzza, and R. Siegwart, "Monocular-slam–based navigation for autonomous micro helicopters in gps-denied environments," *Journal of Field Robotics*, vol. 28, no. 6, pp. 854–874, 2011.

[24] N. Kim, O. Kwon, H. Yoo, Y. Choi, J. Park, and S. Oh, "Topological Semantic Graph Memory for Image Goal Navigation," in *CoRL*, 2022.

[25] D. Shah and S. Levine, "Viking: Vision-based kilometer-scale navigation with geographic hints," *arXiv preprint arXiv:2202.11271*, 2022.

[26] D. Shah, B. Osinski, B. Ichter, and S. Levine, "LM-nav: Robotic navigation with large pre-trained models of language, vision, and action," in *6th Annual Conference on Robot Learning*, 2022. [Online]. Available: https://openreview.net/forum?id=UW5A3SweAH

[27] A. Sridhar, D. Shah, C. Glossop, and S. Levine, "Nomad: Goal masked diffusion policies for navigation and exploration," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 63–70.

[28] Y. Li and J. Košecka, "Learning view and target invariant visual servoing for navigation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 658–664.

[29] X. Meng, N. Ratliff, Y. Xiang, and D. Fox, "Scaling local control to large-scale topological navigation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 672–678.

[30] K. Ehsani, T. Gupta, R. Hendrix, J. Salvador, L. Weihs, K.-H. Zeng, K. P. Singh, Y. Kim, W. Han, A. Herrasti, *et al.*, "Spoc: Imitating shortest paths in simulation enables effective navigation and manipulation in the real world," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 238–16 250.

[31] S. Feng, Z. Wu, Y. Zhao, and P. A. Vela, "Trajectory servoing: Image-based trajectory tracking using slam." *CoRR*, 2021.

[32] S. R. Bista, P. R. Giordano, and F. Chaumette, "Appearance-based indoor navigation by ibvs using line segments," *IEEE robotics and automation letters*, vol. 1, no. 1, pp. 423–430, 2016.

[33] Y. Mezouar and F. Chaumette, "Path planning for robust image-based control," *IEEE transactions on robotics and automation*, vol. 18, no. 4, pp. 534–549, 2002.

[34] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE transactions on robotics and automation*, vol. 12, no. 5, pp. 651–670, 1996.

[35] A. Cherubini, F. Chaumette, and G. Oriolo, "Visual servoing for path reaching with nonholonomic robots," *Robotica*, vol. 29, no. 7, pp. 1037–1048, 2011.

[36] A. Ahmadi, L. Nardi, N. Chebrolu, and C. Stachniss, "Visual servoing-based navigation for monitoring row-crop fields," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4920–4926.

[37] A. Remazeilles, F. Chaumette, and P. Gros, "3d navigation based on a visual memory," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE, 2006, pp. 2719–2725.

[38] A. Diosi, S. Segvic, A. Remazeilles, and F. Chaumette, "Experimental evaluation of autonomous driving based on visual memory and image-based visual servoing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 870–883, 2011.

[39] G. Blanc, Y. Mezouar, and P. Martinet, "Indoor navigation of a wheeled mobile robot along visual routes," in *Proceedings of the 2005 IEEE international conference on robotics and automation*. IEEE, 2005, pp. 3354–3359.

[40] T. Chen, S. Gupta, and A. Gupta, "Learning exploration policies for navigation," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/pdf?id=SyMWn05F7

[41] A. Wahid, A. Stone, K. Chen, B. Ichter, and A. Toshev, "Learning object-conditioned exploration using distributed soft actor critic," in *Conference on Robot Learning*. PMLR, 2021, pp. 1684–1695.

[42] J. Bruce, N. Sünderhauf, P. Deepmind, London, R. Deepmind, and M. Milford, *Learning Deployable Navigation Policies at Kilometer Scale from a Single Traversal*. [Online]. Available: http://proceedings.mlr.press/v87/bruce18a/bruce18a.pdf

[43] Y. Lee, A. Szot, S.-H. Sun, and J. J. Lim, "Generalizable imitation learning from observation via inferring goal proximity," in *Advances in Neural Information Processing Systems*, A. Beygelzimer,

Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021. [Online]. Available: https://openreview.net/forum?id=lp9foO8AFoD

[44] R. Ramrakhya, D. Batra, E. Wijmans, and A. Das, "Pirlnav: Pretraining with imitation and rl finetuning for objectnav," *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2023. [Online]. Available: http://dx.doi.org/10.1109/CVPR52729.2023.01716

[45] C. Zhang, R. Guo, W. Zeng, Y. Xiong, B. Dai, R. Hu, M. Ren, and R. Urtasun, "Rethinking closed-loop training for autonomous driving," in *European Conference on Computer Vision*. Springer, 2022, pp. 264–282.

[46] Y. Hu, J. Yang, L. Chen, K. Li, C. Sima, X. Zhu, S. Chai, S. Du, T. Lin, W. Wang, *et al.*, "Planning-oriented autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 853–17 862.

[47] S. Jung, J. Lee, X. Meng, B. Boots, and A. Lambert, "V-strong: Visual self-supervised traversability learning for off-road navigation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 1766–1773.

[48] G. Loianno and D. Scaramuzza, "Special issue on future challenges and opportunities in vision-based drone navigation." *Journal of Field Robotics*, vol. 37, no. 4, 2020.

[49] S. Krishnan, B. Boroujerdian, W. Fu, A. Faust, and V. J. Reddi, "Air learning: a deep reinforcement learning gym for autonomous aerial robot visual navigation," *Machine Learning*, vol. 110, no. 9, pp. 2501–2540, 2021.

[50] R. Bonatti, R. Madaan, V. Vineet, S. Scherer, and A. Kapoor, "Learning visuomotor policies for aerial navigation using cross-modal representations," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 1637–1644.

[51] A. Majumdar, G. Aggarwal, B. Devnani, J. Hoffman, and D. Batra, "Zson: Zero-shot object-goal navigation using multimodal goal embeddings," *Advances in Neural Information Processing Systems*, vol. 35, pp. 32 340–32 352, 2022.

[52] S. Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, and S. Song, "Cows on pasture: Baselines and benchmarks for language-driven zero-shot object navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 23 171–23 181.

[53] M. Chang, T. Gervet, M. Khanna, S. Yenamandra, D. Shah, S. Y. Min, K. Shah, C. Paxton, S. Gupta, D. Batra, *et al.*, "Goat: Go to any thing," *arXiv preprint arXiv:2311.06430*, 2023.

[54] M. Khanna, R. Ramrakhya, G. Chhablani, S. Yenamandra, T. Gervet, M. Chang, Z. Kira, D. S. Chaplot, D. Batra, and R. Mottaghi, "Goat-bench: A benchmark for multi-modal lifelong navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 373–16 383.

[55] G. Zhou, Y. Hong, and Q. Wu, "Navgpt: Explicit reasoning in vision-and-language navigation with large language models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 7, 2024, pp. 7641–7649.

[56] G. Zhou, Y. Hong, Z. Wang, X. E. Wang, and Q. Wu, "Navgpt-2: Unleashing navigational reasoning capability for large vision-language models," *arXiv preprint arXiv:2407.12366*, 2024.

[57] J. Chen, B. Lin, R. Xu, Z. Chai, X. Liang, and K.-Y. K. Wong, "Mapgpt: Map-guided prompting for unified vision-and-language navigation," *arXiv preprint arXiv:2401.07314*, 2024.

[58] H. Wang, J. Qin, A. Bastola, X. Chen, J. Suchanek, Z. Gong, and A. Razi, "Visiongpt: Llm-assisted real-time anomaly detection for safe visual navigation," *arXiv preprint arXiv:2403.12415*, 2024.

[59] J. Zhang, K. Wang, R. Xu, G. Zhou, Y. Hong, X. Fang, Q. Wu, Z. Zhang, and W. He, "Navid: Video-based vlm plans the next step for vision-and-language navigation," *arXiv preprint arXiv:2402.15852*, 2024.

[60] H.-T. L. Chiang, Z. Xu, Z. Fu, M. G. Jacob, T. Zhang, T.-W. E. Lee, W. Yu, C. Schenck, D. Rendleman, D. Shah, *et al.*, "Mobility vla: Multimodal instruction navigation with long-context vlms and topological graphs," *arXiv preprint arXiv:2407.07775*, 2024.

[61] S. Nasiriany, F. Xia, W. Yu, T. Xiao, J. Liang, I. Dasgupta, A. Xie, D. Driess, A. Wahid, Z. Xu, *et al.*, "Pivot: Iterative visual prompting elicits actionable knowledge for vlms," *arXiv preprint arXiv:2402.07872*, 2024.

[62] A. J. Sathyamoorthy, K. Weerakoon, M. Elnoor, A. Zore, B. Ichter, F. Xia, J. Tan, W. Yu, and D. Manocha, "Convoi: Context-aware navigation using vision language models in outdoor and indoor environments," *arXiv preprint arXiv:2403.15637*, 2024.

[63] P. Furgale and T. D. Barfoot, "Visual teach and repeat for long-range rover autonomy," *Journal of field robotics*, vol. 27, no. 5, pp. 534–560, 2010.

[64] S. Šegvić, A. Remazeilles, A. Diosi, and F. Chaumette, "A mapping and localization framework for scalable appearance-based navigation," *Computer Vision and Image Understanding*, vol. 113, no. 2, pp. 172–187, 2009.

[65] A. M. Zhang and L. Kleeman, "Robust appearance based visual route following for navigation in large-scale outdoor environments," *The International Journal of Robotics Research*, vol. 28, no. 3, pp. 331–356, 2009.

[66] D. Dall'Osto, T. Fischer, and M. Milford, "Fast and robust bio-inspired teach and repeat navigation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 500–507.

[67] M. Mattamala, N. Chebrolu, and M. Fallon, "An efficient locally reactive controller for safe navigation in visual teach and repeat missions," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2353–2360, 2022.

[68] T. Krajník, F. Majer, L. Halodová, and T. Vintr, "Navigation without localisation: reliable teach and repeat based on the convergence theorem," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1657–1664.

[69] L. Halodová, E. Dvořráková, F. Majer, T. Vintr, O. M. Mozos, F. Dayoub, and T. Krajník, "Predictive and adaptive maps for long-term visual navigation in changing environments," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 7033–7039.

[70] T. Do, L. C. Carrillo-Arce, and S. I. Roumeliotis, "High-speed autonomous quadrotor navigation through visual and inertial paths," *The International Journal of Robotics Research*, vol. 38, no. 4, pp. 486–504, 2019.

[71] T. Krajník, P. Cristóforis, K. Kusumam, P. Neubert, and T. Duckett, "Image features for visual teach-and-repeat navigation in changing environments," *Robotics and Autonomous Systems*, vol. 88, pp. 127–141, 2017.

[72] S. Levine and D. Shah, "Learning robotic navigation from experience: principles, methods and recent results," *Philosophical Transactions of the Royal Society B*, vol. 378, no. 1869, p. 20210447, 2023.

[73] D. Shah, A. Sridhar, N. Dashora, K. Stachowicz, K. Black, N. Hirose, and S. Levine, "Vint: A large-scale, multi-task visual navigation backbone with cross-robot generalization," in *7th Annual Conference on Robot Learning*, 2023.

[74] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 224–236.

[75] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, "LightGlue: Local Feature Matching at Light Speed," in *ICCV*, 2023.

[76] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[77] S. K. Ramakrishnan, A. Gokaslan, E. Wijmans, O. Maksymets, A. Clegg, J. Turner, E. Undersander, W. Galuba, A. Westbury, A. X. Chang, M. Savva, Y. Zhao, and D. Batra, "Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai," *ArXiv*, vol. abs/2109.08238, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:237563216

[78] J. Krantz, S. Lee, J. Malik, D. Batra, and D. S. Chaplot, "Instance-specific image goal navigation: Training embodied agents to find object instances," *arXiv preprint arXiv:2211.15876*, 2022.

[79] K. Yadav, J. Krantz, R. Ramrakhya, S. K. Ramakrishnan, J. Yang, A. Wang, J. Turner, A. Gokaslan, V.-P. Berges, R. Mootaghi, O. Maksymets, A. X. Chang, M. Savva, A. Clegg, D. S. Chaplot, and D. Batra, "Habitat challenge 2023," https://aihabitat.org/challenge/2023/, 2023.