2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Efficient Terrain Map Using Planar Regions for Footstep Planning on Humanoid Robots

Bhavyansh Mishra[1,2], Duncan Calvert[1,2], Sylvain Bertrand[2], Jerry Pratt[1,2], Hakki Erhan Sevil[1], and Robert Griffin[1,2]

***Abstract*— Humanoid robots possess the ability to perform complex tasks in challenging environments. However, they require a model of the surroundings in a representation that is sufficient enough for downstream tasks such as footstep planning. The maps generated by existing mapping algorithms are either sparse, insufficient for footstep planning, memory intensive, or too slow for dynamic humanoid behaviors. In this work, we develop a mapping algorithm that combines planar region measurements along with kinematic-inertial state estimates to build a dense but efficient map of bounded planar surfaces. We present novel algorithms for plane feature matching, tracking and registration for mapping within a factor graph framework. The generated map is not only memory efficient, but also offers higher reliability and speed in bipedal footstep planning, than was possible earlier. The complete algorithm is also demonstrated using a full-scale humanoid robot, Nadia, walking over both flat ground and rough terrain utilizing the generated terrain map.**

I. Introduction

Humanoid robots are capable of executing highly complex tasks and operations that are commonly performed by human beings [1]. However, being able to achieve such tasks robustly in various environments with challenging terrain and time-critical conditions is still an open area of research, partly due to the limitations in being able to perceive the environment sufficiently [2]. Disaster response, space exploration, warehouse automation, and building construction are just a few examples from the vast space of potential use cases for humanoid robots [3], [4]. Tremendous research over the past decades on humanoids has vitalized the need for reliable representations of both the terrain and the surroundings. For performing fast behaviors on legged robots, the challenge of rapidly processing data from on-board sensors to extract both a model of the environment and the robot's location at high rates must be addressed. In this work, we present a mapping algorithm that uses kinematic, inertial, and planar region measurements to generate a consistent model of the terrain for bipedal humanoid locomotion (as shown in Figure 1). Our approach exploits the parallelism of GPU kernels, compact representation of the environment as planar regions, as well as incremental smoothing for factor graph optimization to achieve high update rates, and generate a memory-efficient terrain map. Most existing approaches on simultaneous localization and mapping (SLAM) utilize low-level features such as points, edges, blobs, and surfels to populate sparse or dense maps. The proposed approach generates a dense but highly compressed map representation using planar regions for footstep planning and traversability assessment on legged robots. Our map representation also reduces on-board resource usage, and allows for efficient map storage as well as map sharing for multi-agent and distributed systems. The primary contributions of this work are:

1) Novel approach for planar region feature representation, data association, and feature boundary merging, using depth maps from solid-state LIDARs.
2) A fast and novel plane registration algorithm termed Iterative Quaternion Averaging (IQA) which exploits the low-dimensional representation of planar surfaces and their relative orientations to extract the rigid body transform between planar region sets.
3) Planar region based map that tracks boundaries of surfaces and provides a stable and noise-free model of the terrain for fast footstep planning for dynamic locomotion on humanoid robots.
4) A factor-graph based formulation of the mapping problem with kinematic-inertial and planar region odometry and landmark measurements.

The paper first describes the extraction, data association, and registration for planar region measurements, and then presents the factor graph based framework for combining

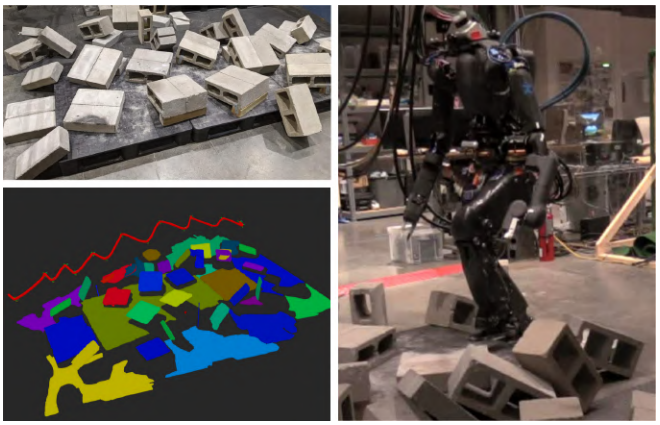

Fig. 1. Terrain map generated using using GPU-extracted planar regions (bottom-left) over rough cinder block field (top-left) in real-time at 50-100 Hz. The map tracks boundaries for the planar regions as they grow due to camera motion. We evaluate the map of planar regions for footstep planning and rough terrain locomotion on the Nadia robot (right).

[1]Author is with the University of West Florida, 11000 University Pkwy, Pensacola, FL 32514, United States
[2]Author is with the Florida Institute for Human and Machine Cognition, 40 S Alcaniz St, Pensacola, FL 32502, United States
Email : {bmishra,dcalvert,sbertrand,jpratt,rgriffin}@ihmc.org,hsevil@uwf.edu

plane landmark measurements along with kinematic-inertial state estimates using fast incremental smoothing. The paper then describes the post-optimization merging of planar region surfaces and boundaries for extended mapping. Finally, it analyzes the effectiveness of the approach for walking experiments on a full-scale (29 DoF) humanoid robot Nadia, and presents quantitative comparisons for localization accuracy, footstep planning performance on the generated map, as well as time and memory usage.

II. Related Work

Most existing works such as SLAM algorithms have focused on localization accuracy, and consequently, map representation and utility have remained unexplored [5], [6]. However, legged and manipulation systems require reliable models of the world for contact-critical tasks such as walking, running, manipulation, etc. For the purposes of humanoid terrain perception in various environments and conditions, existing methods for mapping are either too slow for dynamic humanoid behaviors [7], [8], too sparse for tasks such as footstep planning [9]–[13], compute and memory intensive [14]–[16], leverage environment specific assumptions (such as the Manhattan World, fiducial markers, existence of a ground plane) [8], [17], [18], or have only been demonstrated to function on statically stable robots [19]. Elevation maps have also been utilized for both quadrupedal [20]–[22] and bipedal [23] legged robots. However, elevation maps degrade due to drift in state estimation, and also fail to represent hanging structures. Techniques for reducing state estimation drift for legged robots have also been presented [24], [25]. Our previous work [26], [27] extracted local planar regions and directly used them for footstep planning. However, this severely limited the planning due to noisy detections, self-occlusions, and limited field-of-view (described in Figure 2). Past approaches have also forayed into localization and mapping using planar structures. Dense planar SLAM [28] employed dense ICP on the GPU, but did not use plane landmarks in the global optimization. Roychoudhury et al. [18] developed a 3D polygonal mapping approach using planar surface registration and merging for environments where a ground plane could be estimated. Their approach was only evaluated over flat-ground walking in indoor environment using a Nao humanoid robot. Hsiao et al. [29] employed dense and semi-dense RGBD odometry to perform keyframe based dense SLAM using planes as landmarks. Their approach relied on color information which is limited in unfavorable lighting conditions and texture-less environments. Trevor et al. [19] presented a planar surface SLAM in which they inserted planar feature measurements as constraints in the factor graph. However, their experiments were performed on statically stable robots moving on flat-ground in a structured environment. Kanoulas et al. [30] approximated the terrain using ten different curved patch types to balance between expressiveness and compactness of representation. Learning-based approaches for terrain modelling have been limited by either the simulation environment or the dataset that was used during training [31], [32]. A description of the various types of terrain that humanoid and legged robots need to be able to perceive and traverse was provided in the thesis by Stumpf [2]. This paper presents a fast and reliable mapping algorithm for the types of terrain that possess a reasonable number of arbitrarily oriented planar surfaces in environments in which humanoid robots are expected to operate. Our approach does degrade in highly unstructured environments, but delivers a reliable approach for locomotion in both semi-structured and urban environments.

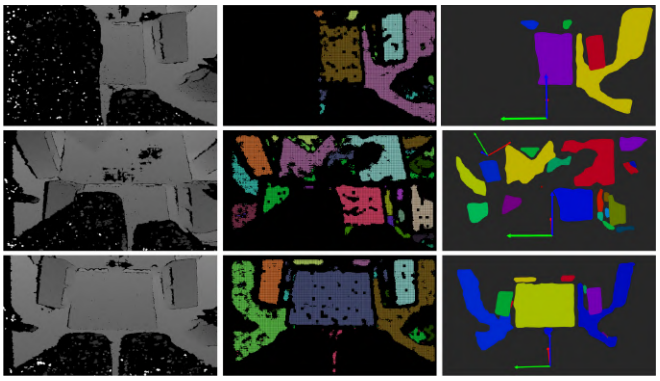

Fig. 2. Instantaneous local rapid regions generated by [33] are insufficient for planning footsteps either towards the front, back, or sides of the robot, due to major occlusions by swing foot (top), noisy and broken planar regions (middle), and inability to observe the terrain under the robot knees (bottom). However, a persistent global map does not suffer from such problems and allows for footstep planning with significantly more reliability, frequency, and feasibility.

III. Approach

At every frame, we first extract a set of planar regions using our GPU-accelerated algorithm [33]. The set $\mathcal{P}_S := \{\pi_1, \pi_2, ...\pi_N\}$ consists of planar regions $\pi_i := \{\mathbf{p}_i, \Omega_i, {}^S\mathbf{T}_L\}$ in the sensor frame $\mathcal{S}$. Here, $\mathbf{p}_i$ is the vector of plane parameters, $\Omega_i := \{\mathbf{v}_1, \mathbf{v}_2, ...\mathbf{v}_H\}$ is the concave hull of 2D points $\mathbf{v}_k \in \mathbb{R}^2$ representing the boundary in the local frame $\mathcal{L}$ (within the plane), and ${}^{\mathcal{S}}\mathbf{T}_{\mathcal{L}} \in$ SE(3) is the transform to go from the local frame $\mathcal{L}$ to the sensor frame $\mathcal{S}$. The local frame $\mathcal{L}$ is chosen with an arbitrary yaw and XY-plane exactly along the planar region surface with $\hat{Z}$ along the surface normal of the planar region. For transformations in three-dimensions, we elevate the 2D points into the local frame $\mathcal{L}$ by adding a zero Z dimension. Our algorithm also extracts two different odometry estimates along with these planar regions, and ultimately performs a factor graph based smoothing over all such constraints to find the optimal localization and map estimates which can then be used by the robot (as shown in Figure 3).

A. Iterative Quaternion Averaging for Plane Registration

Terrain mapping with accurate boundaries requires precise registration of the planar regions to previously observed regions in the map. We perform a novel Iterative Quaternion Averaging (IQA) based registration to compute the rotation and translation of the current region set $\mathcal{P}_i$ to the previous keyframe set $\mathcal{P}_{k-1}$. The algorithm leverages quaternion averaging to rapidly estimate the sensor transform ${}^{k-1}\mathbf{T}_i$ to

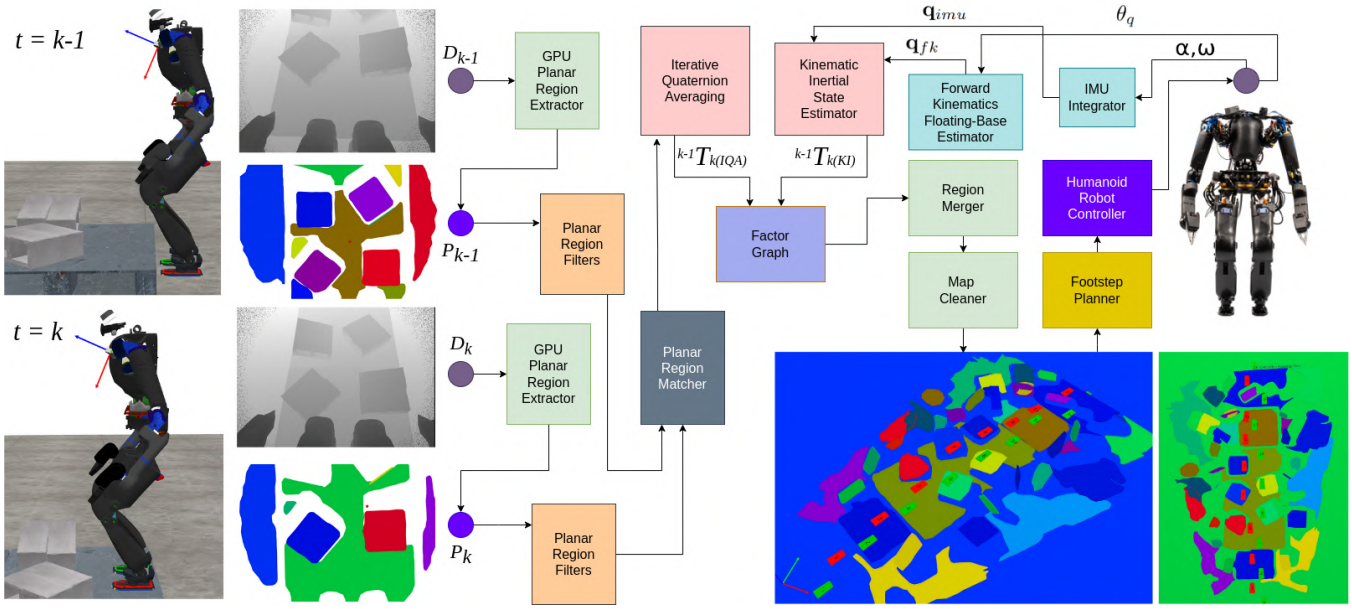

Fig. 3. Planar region set from the latest keyframe at $t = k$ (extracted from depth map D_k) is registered against the set from previous keyframe $t = k-1$ (extracted from depth map D_{k-1}) to obtain the IQA odometry. The kinematic-inertial odometry estimate is computed using the IMU and encoder sensors on the robot. All odometry and landmark measurements are inserted as constraints into the factor graph. The final map is used for planning footsteps for locomotion.

the previous keyframe which minimizes the plane registration error. At every frame, the current local set $\mathcal{P}_i$ is first matched against the set from previous keyframe $\mathcal{P}_{k-1}$ based on a similarity score combining both the angular and distance metrics between two regions as,

$$\sigma(\pi_i, \pi_j) = W_a \hat{\mathbf{n}}_i \cdot \hat{\mathbf{n}}_j + W_b(\mathbf{p}_i - \mathbf{p}_j) \cdot \hat{\mathbf{n}}_i + W_c \beta(\Omega_i, \Omega_j), \quad (1)$$

where (W_a, W_b, W_c) are hand-tuned weights and $(\mathbf{p}, \hat{\mathbf{n}})$ are the origin and surface normal of the planar region. We threshold the similarity score between each pair by the parameter α_{sim} to get the set of usable matching pairs $\mathcal{M} := \{(m, n) \mid \pi_m \in \|\mathcal{P}_i\|, \pi_n \in \|\mathcal{P}_{k-1}\|\}$. The function $\beta(\Omega_i, \Omega_j)$ computes the Intersection-over-Union (IoU) of the world frame 3D bounding boxes for given planar region pair (π_i, π_j). The score values for the successful matches are also cached in the memory for weighting the registration calculation later on.

For each matching pair (m, n), we compute the candidate transform $^{k-1}\mathbf{T}_i$ based on the unit quaternion $\mathbf{q}_{m,n}$ (rotation between plane normals) and vector $\mathbf{t}_{m,n}$ (displacement between region origins) that aligns only that specific planar region pair optimally. We then compute an evaluation score $\Delta(^{k-1}\mathbf{T}_i, \mathcal{P}_i, \mathcal{P}_{k-1})$ for this candidate transform based on its fitness to register the entire region set $\mathcal{P}_i$ to the previous keyframe set $\mathcal{P}_{k-1}$. The evaluation score is computed as the sum of similarity scores for all the matching pairs between $\mathcal{P}_{k-1}$ and $^{k-1}\mathbf{T}_i(\mathcal{P}_i)$ weighted by the average surface area of the source-target region pair $\hat{A}(\pi_m, \pi_n)$ as,

$$\Delta(^{k-1}\mathbf{T}_i, \mathcal{P}_i, \mathcal{P}_{k-1}) = \sum_{\pi_m, \pi_n}^{\mathcal{P}_i, \mathcal{P}_{k-1}} \sigma(\pi_m, \pi_n) * \hat{A}(\pi_m, \pi_n), \quad (2)$$

where $\hat{A}(\pi_i, \pi_j) = (A(\pi_i) + A(\pi_j))/2$ and $A(\pi)$ computes the metric surface area of the planar region π. This ensures that the planar regions with larger surface areas have higher influence on the overall score. The evaluation score is then used as the weight for the candidate transform $(^{k-1}\mathbf{q}_i, ^{k-1}\mathbf{t}_i)$ and arranged in a diagonal matrix of weights $\mathbf{W}$ as,

$$\mathbf{W} = \begin{bmatrix} w_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & w_M \end{bmatrix} \quad (3)$$

This matrix of weights contains the evaluation score each individual pair-wise candidate quaternion estimates. We then create the matrix $\mathbf{Q}$ of all the pairwise candidate quaternion estimates $\mathbf{q}_{i,j}$ stacked as,

$$\mathbf{Q} = \begin{pmatrix} | & | & & | \\ \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_M \\ | & | & & | \end{pmatrix}. \quad (4)$$

The final estimate $\mathbf{q}^*$ that minimizes the similarity cost for all matching pairs combined is then computed using a singular value decomposition of the matrix $\mathbf{QWQ}^T$,

$$\mathbf{QWQ}^T = \mathbf{U\Sigma V}^T \quad (5)$$

wherein, $\mathbf{q}^*$ is obtained by extracting the singular vector $\mathbf{q}^* = \mathbf{u}_{max}$ from $\mathbf{U}$ corresponding to the largest singular value in the diagonal of the $\mathbf{\Sigma}$ matrix. The weight matrix allows the more representative quaternion estimate to be weighted higher than the rest. Once the optimal unit quaternion $\mathbf{q}^*$ has been estimated, the rotation $\mathbf{R}(\mathbf{q}^*) \in \mathrm{SO}(3)$ is applied to all the individual regions in the source set (vertices elevated to three-dimensions). After this rotation, the optimal translation can be estimated as the vector $\mathbf{t}^* \in \mathbb{R}^3$ that minimizes $g(\mathbf{t}) = \sum_{(m,n)}^{M} (\mathbf{t}_n - (\mathbf{t}_m + \mathbf{t}) \cdot \hat{\mathbf{n}})^2$, where $\mathbf{t}_m$ is the source set region origin and $\mathbf{t}_n$ is the target set region origin for the matching region pair (m, n). The final

odometry estimate at the next keyframe $t = k$ is then given by ${}^{k}\mathbf{T}_{k-1,IQA} = \mathcal{T}({}^{k-1}\mathbf{q}_k, {}^{k-1}\mathbf{t}_k)$, where the operator $\mathcal{T}(\theta, \tau)$ generates the SE(3) pose corresponding to rotation by quaternion θ, following by a translation τ.

The algorithm performs an alternating procedure of matching and averaging iteratively until the reduction ratio between consecutive errors falls below a hand-tuned threshold value. The error used is the sum of the similarity metric for all matching pairs for that iteration. The weights in the matrix $\mathbf{W}$ ensure that the contribution by wrong quaternion estimates affect the solution by a smaller magnitude, which makes the algorithm robust to outlier matching pairs and erroneously detected planar regions due to noise in the feature extraction phase. Overall, this approach works well in various types of environments that have at least 3-4 unique planar surfaces to detect. In cases where this assumption is broken, the robot odometry is trusted more, which maintains a fairly high accuracy in both the localization and map estimates.

B. Kinematic-Inertial State Estimator

The odometry from high-frequency proprioceptive sensors such as IMUs and encoders is integrated and provided to the factor graph for initialization and additional pose-pose constraints between keyframes $(k-1, k)$. Given the joint angles θ_q and initial foot poses $({}^{w}\mathbf{T}_l, {}^{w}\mathbf{T}_r)$, we compute the forward kinematics based floating-base pose $(\mathbf{q}_{fk}, \mathbf{t}_{fk})$ by averaging the contribution of each foot weighted by the loading force on the foot. During single support the weight on the swing foot is fixed to zero. The forward kinematic estimate for the floating-base quaternion is computed by the weighted spherical linear interpolation between the left and right estimates as

$$\mathbf{q}_{fk} = \mathbf{q}_r(\mathbf{q}_r^{-1}\mathbf{q}_l)^{\frac{f_r}{f_l+f_r}}, \tag{6}$$

where f_l and f_r are the magnitude of normal forces on the left foot and right foot, respectively. We also use the angular velocity measurements $\omega \in \mathbb{R}^3$ generated by the IMU to compute another estimate as the quaternion composition of $\mathbf{q}_{k-1}$ and $\mathbf{q}_{\omega\delta t}$,

$$\mathbf{q}_{imu} = \mathbf{q}_{k-1} \otimes \mathbf{q}_{\omega\delta t}, \tag{7}$$

where $\mathbf{q}_{\omega\delta t} = [\cos(\delta t/2), \omega \sin(\delta t/2)]^T$ is the quaternion representation for the axis-angle rotation $(\omega, \delta t)$. Finally, the overall orientation estimated is computed by the interpolation between $\mathbf{q}_{imu}$ and $\mathbf{q}_{fk}$ as

$$\mathbf{q}_k = \mathbf{q}_{fk}(\mathbf{q}_{fk}^{-1}\mathbf{q}_{imu})^{\alpha}, \tag{8}$$

where α is a hand-tuned weight to balance between IMU drift and kinematic noise. For linear position and velocity estimates, we use a standard Kalman filter to fuse both linear acceleration and kinematics-based linear estimates.

The kinematic-inertial floating-base estimate is computed as $\hat{\mathbf{X}}_{KI} = \mathcal{T}(\mathbf{q}_k, \mathbf{t}_k)$. This final kinematic-inertial estimate is then supplied to the factor graph for the overall global optimization along with landmark measurements and the IQA estimate at every keyframe.

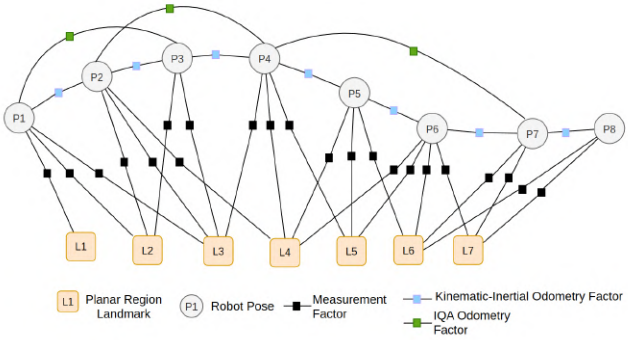

Fig. 4. Factor graph for terrain mapping using kinematic-inertial odometry, IQA odometry, and planar region measurement constraints. Oriented plane factors are used to add 4-DoF constraints for the plane parameters for unique planar surfaces in the scene, and pose-pose constraints are taken from both IQA and kinematic-inertial estimator (integrated keyframe-to-keyframe).

C. Factor Graph Optimization for Landmarks and Poses

We model the mapping problem as a factor graph of poses, landmarks, and the probabilistic constraints between them imposed by noisy measurements. The front-end consisting of planar region extraction, matching, and registration, along with kinematic-inertial odometry measurements generates the constraints to be inserted into the factor graph. Specifically, planar region measurement constraints are inserted as 4 DoF oriented plane landmark factors. The odometry measurements from both our plane-based IQA algorithm, as well as the robot state estimator are inserted as odometry factors between two successive sensor poses at keyframes. The keyframes are selected based on a thresholding criterion on both rotation and translation of the sensor as described earlier. The posterior probability for the factor graph in Figure 4 of the variables $\mathcal{X}$ consisting of poses $\mathbf{x}_k$ and landmarks $\mathbf{l}_m$, given the measurements $\mathcal{Z}$ consisting of IQA odometry estimates $\mathcal{T}_k^{IQA}$ and kinematic-inertial odometry estimates $\mathcal{T}_k^{KI}$, as well as planar region measurements $\mathcal{L}_m$ up to $t = k$ is then given by:

$$\begin{aligned} P(\mathcal{X}|\mathcal{Z}) = &\prod_k P(\mathcal{T}_k^{IQA}|\mathbf{x}_k, \mathbf{x}_{k-1}) \\ &* \prod_k P(\mathcal{T}_k^{KI}|\mathbf{x}_k, \mathbf{x}_{k-1}) \\ &* \prod_m P(\mathcal{L}_m|\mathbf{x}_k, \mathbf{l}_m). \end{aligned} \tag{9}$$

The negative log-posterior can then be minimized to obtain the optimal solution $\mathcal{X}^*$ by solving the following non-linear least squares problem,

$$\begin{aligned} \mathcal{X}^* = &\arg\min_{\mathbf{x}_i} \sum_k \|\mathbf{r}_{ki}\|^2_{\Sigma_{KI}} \\ &+ \sum_k \|\mathbf{r}_{iqa}\|^2_{\Sigma_{IQA}} + \sum_k \|\mathbf{r}_m\|^2_{\Sigma_M}, \end{aligned} \tag{10}$$

where $\mathbf{r}_m, \mathbf{r}_{iqa}, \mathbf{r}_{ki}$ are the residual functions for the measurement models for plane landmarks, IQA odometry and kinematic-inertial odometry, respectively. For the plane landmarks, we employ the measurement function definition as in [19]. The maximum-a-posteriori estimate for the factor graph contains the optimal keyframe poses $\mathbf{x}_i^*$ and planar landmark coefficients π_i^*. The non-linear least squares is optimized using the Incremental Smoothing and Mapping (iSAM2 [34])

algorithm to generate real-time solutions for poses and planar landmarks.

D. Planar Region Association, Merging, and Clean-Up

Despite the efficiency benefits, our use of the planar region features is inherently more challenging for localization and mapping, as it requires careful identification of cross-set and self-set matching correspondences of regions that eventually require merging after factor graph optimization. Unlike visual keypoints and other geometric features used in the literature, these bounded planar regions can grow and shrink their boundaries, as well as merge with other regions to create fused regions. This problem motivated the development of a novel cross-and-self region match checking algorithm that could search exhaustively for feature correspondences and find most edge cases for merging of planar regions from two different sets for accurate final map region boundaries. Our match checking algorithm considers two planar region sets π_a and π_b, and uses three nested loops to go through all combinations of feasible mergeable pairs, and even identifies edge cases such as bridge regions. This checks to ensure that the IDs of matching regions remain consistent throughout the life of the algorithm, and that only the unique planar surfaces are inserted as factors into the factor graph.

On a separate thread, all map regions that were inserted within a time window of τ_{kf} from the current keyframe, and were not observed for at least η_{kf} times, are removed from the map. This ensures that incorrectly detected planar region features are discarded before they can corrupt the factor graph by generating undesirable factors. All pairs of regions that intersect at an angle γ in the range $\pi/2+\varepsilon < \gamma < \pi/2-\varepsilon$ are chopped off by removing the external and minor parts at the intersecting edges. This ensures clean boundaries in the map, and improves the chances of finding correct data association for the planar regions.

IV. Experiments and Results

In this section, we demonstrate the application of our novel mapping algorithm for fast and dynamic bipedal walking experiments on rough terrain using the IHMC and Boardwalk Robotics Nadia robot (full humanoid robot with 29 degrees-of-freedom) [35]. The algorithm was evaluated with experiments making Nadia go over flat ground, rough cinder block field, and mock staircase. In all the experiments, signals from the kinematic, inertial, and depth sensors were used to compute the factor-graph optimization on an on-board highly integrated system with Intel i7 and NVidia GTX 1050Ti. The depth maps were obtained from a D455 mounted on the chest of the robot facing down on the terrain (as shown in Fig. 3). The experiments were performed in both sunlit and nighttime conditions in the IHMC Robotics Lab with glass windows for ambient sunlight to pass in. The lab area is equipped with 12 OptiTrack motion capture cameras to track various rigid bodies in the covered area. In the rough terrain walking sequences, the robot first walks in positive-X direction in map frame (E1), and then walks back in the negative-X direction (E2) through the same rough terrain of a random

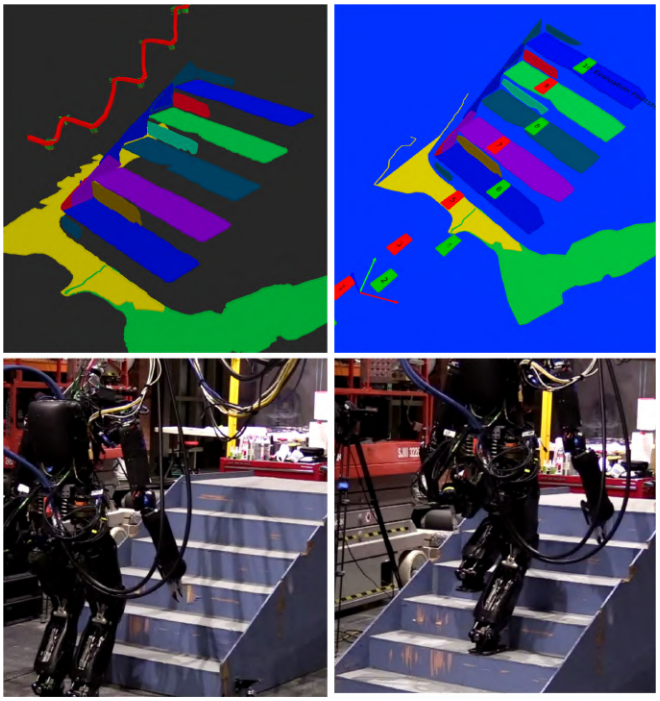

Fig. 5. Footstep planning on the planar region map to climb up stairs. The final map facilitates clean footstep planning unaffected by self-occlusions and flickering nature of instantaneously extracted planar regions from any specific frame of data.

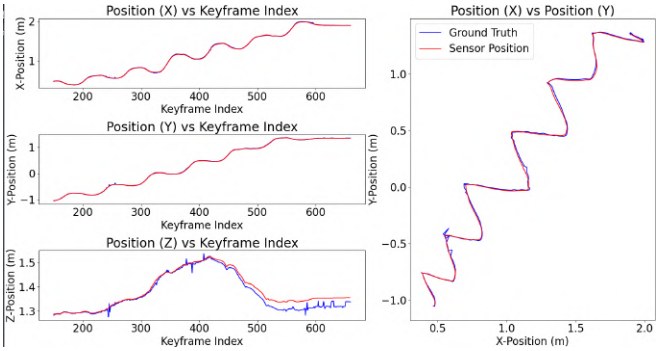

Fig. 6. Sensor localization comparison against motion capture based ground truth trajectory for sequence E1

cinder block field. In the flat ground walking sequences (E3, E4), the robot is made to traverse through a relatively narrow turn on flat-ground through between piles of cinder blocks on the sides (used as obstacles). For walking, a swing time of 1.2s and a transfer time of 0.6s was used for relatively fast but stable locomotion over both flat ground and rough terrain. We compute the Absolute Trajectory Error (ATE) as the root-mean-square error (RMSE) for absolute position error in the map frame to compare the localization estimate against the motion-capture based ground truth trajectory (shown in Figure 6 for E1) as,

$$\varepsilon_{ATE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} |\mathbf{p}_{e,i} - \mathbf{p}_{g,i}|^2}, \tag{11}$$

where $|.|$ denotes the Euclidean norm, $\mathbf{p}_{e,i}$ is the optimal position estimate from our algorithm, and $\mathbf{p}_{g,i}$ is the ground

TABLE I

ABSOLUTE TRAJECTORY ERROR (ATE), MAP STATISTICS AND FOOTSTEP PLANNING RESULTS FOR WALKING EXPERIMENTS (E1-E4)

Global Mapping							Footstep Planning							
Map Statistics							Local (Front)		Map (Front)		Local (Back)		Map (Back)	
ID	ATE	Area	Regions	Vertices	Time	Scans	SR	Time	SR	Time	SR	Time	SR	Time
E1	0.018m	9.28m^2	39	4209	0.024s	500	0.29	0.045s	**0.41**	**0.039s**	0.0	0.5s	**0.46**	**0.081s**
E2	0.063m	9.55m^2	27	2953	0.043s	568	0.235	**0.02s**	**0.38**	0.06s	0.0	0.5s	**0.36**	**0.09s**
E3	0.019m	9.82m^2	12	1721	0.046s	888	0.834	**0.032s**	**0.902**	0.06s	0.0	0.5s	**0.85**	**0.133s**
E4	0.019m	7.26m^2	10	1288	0.041s	868	**0.955**	**0.022s**	0.951	0.071s	0.0	0.5s	**0.957**	**0.095s**

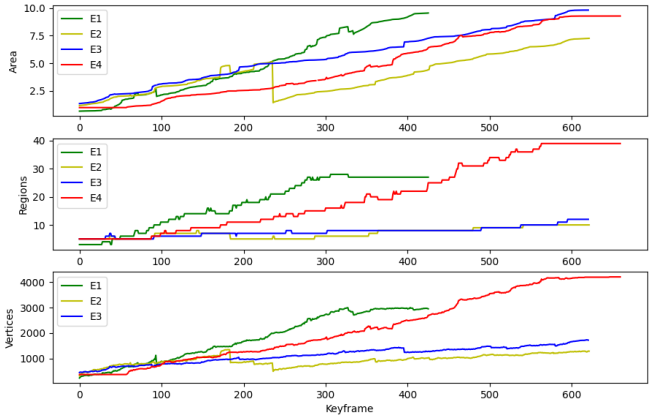

Fig. 7. Map statistics and footstep planning results for experimental sequences E1-E4.

truth position. The ATE errors for the sequences (E1-E4) are shown in the Table I. Highlighting the efficiency of our terrain representation, we further present a comparison of memory consumption of our approach against existing mapping algorithms for the *fr3/office* sequence from the TUM RGB-D benchmarking dataset (shown in the Table II. Due to the compact nature of the planar region definition, the overall memory needed to store and transmit a map of planar regions is significantly lower than volumetric, pointcloud, and other voxel and surfel based representations. The total memory $\mathcal{M}$ (in MB) used was computed using the expression $\mathcal{M} = (\sum_{\pi_i \in P} 8 * |\Omega_i| + 44) * 2^{-20}$, where vertex $\mathbf{v} \in \Omega_i$ takes 12 bytes (three 32-bit floats), as well as the region parameters (position, quaternion and plane coefficients) take 44 bytes (eleven 32-bit floats). Metrics from the final generated map for sequences E1-E4 such as total number of planar regions, total surface area mapped, and total vertex count were also computed to identify the rate of map growth over successive keyframes (provided in Figure 7). Finally, footstep planning success rates (SR)(on table I) for forward and backward walking on both instantaneous local region set and global persistent map were collected. For footstep planning the A* based footstep planner [36] was used. At randomly selected keyframes, the footstep planner was tasked with generating footsteps for a goal position placed (1) forward by 0.5 m on local set, (2) forward by 0.5 m on map set, (3) backward by 0.5 m on local set, and (4) backward by 0.5 m on map set. The results clearly demonstrate improvements in planning footsteps both within and outside the field-of-view (FOV) of the robot. This is mainly because the local instantaneous detections are noisy

TABLE II

MAP SIZE COMPARISON AGAINST MAPS GENERATED BY OTHER EXISTING DENSE ALGORITHMS ON TUM RGB-D BENCHMARK DATASET IN MB

Method	30s	60s	90s	120s	150s	Time
KinectFusion	70.1	98.9	158.2	276.7	436.6	25
BAD SLAM	13.1	50.0	86.4	109.1	123.8	370
ElasticFusion	17.5	29.0	38.2	53.5	63.6	72.6
ORBSLAM2	18.9	21.9	30.4	34.4	38.8	1793.02
RTABMap	8.7	17.0	25.1	33.1	37.9	-
MeSLAM	1.9	1.9	1.9	1.9	1.9	-
Ours	**0.02**	**0.04**	**0.06**	**0.06**	**0.08**	**20.4**

and limited by both the FOV of the sensor and the occlusions due to robot legs. However, the map is significantly less noisy, exists in spaces outside FOV, and eliminates self-occlusions by temporal persistence. This allows the robot to take immediate steps in all directions, which can be critical for operation in challenging environments, such as in Figure 5. We also provide open source access to the code and dataset at [37].

V. CONCLUSION AND FUTURE WORK

Finding the best terrain representation for humanoid robot locomotion in diverse and challenging environments, as well as, in time-critical and adverse conditions, still remains to be an open research problem. However, given the existence of a reasonable number of planar surfaces in the environment, the presented algorithm can generate a globally consistent and compact map of planar regions for reliable and efficient footstep planning on humanoid robots. Novel loop closures for detection and optimization could help extend the work for long-term uses on the robot in the future. The simple geometric planar region feature representation was also found to be prone to incorrect matching. Future works could also explore more unique and expressive feature descriptors that identify and match planar region elements robustly, accurately and consistently in diverse environments. Height maps also offer a promising alternative for terrain representation but their use for humanoid robots in multi-storied buildings and in presence of hanging structures remains to be explored.

VI. ACKNOWLEDGMENT

We would like to acknowledge the contribution by Dexton Anderson for software support needed to achieve this work. Further, this work was partially funded by the Office of Naval Research (ONR Grant Number: N00014-22-1-2593).

REFERENCES

[1] K. Darvish, L. Penco, J. Ramos, R. Cisneros, J. Pratt, E. Yoshida, S. Ivaldi, and D. Pucci, "Teleoperation of humanoid robots: A survey," *IEEE Transactions on Robotics*, pp. 1–22, 2023.

[2] A. Stumpf, "An integrated concept for footstep planning and navigation for different types of multi-legged robots in challenging environments," 2020.

[3] S. J. Jorgensen, M. W. Lanighan, S. S. Bertrand, A. Watson, J. S. Altemus, R. S. Askew, L. Bridgwater, B. Domingue, C. Kendrick, J. Lee, M. Paterson, J. Sanchez, P. Beeson, S. Gee, S. Hart, A. H. Quispe, R. Griffin, I. Lee, S. McCrory, L. Sentis, J. Pratt, and J. S. Mehling, "Deploying the NASA valkyrie humanoid for IED response: An initial approach and evaluation summary," in *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, 2019, pp. 1–8.

[4] R. Tedrake, M. Fallon, S. Karumanchi, S. Kuindersma, M. Antone, T. Schneider, T. Howard, M. Walter, H. Dai, R. Deits, M. Fleder, D. Fourie, R. Hammoud, S. Hemachandra, P. Ilardi, C. Pérez-D'Arpino, S. Pillai, A. Valenzuela, C. Cantu, C. Dolan, I. Evans, S. Jorgensen, J. Kristeller, J. A. Shah, K. Iagnemma, and S. Teller, "A summary of team MIT's approach to the virtual robotics challenge (VRC)," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 2087–2087.

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

[6] B. Mishra, R. Griffin, and H. E. Sevil, "Modelling software architecture for visual simultaneous localization and mapping," *Automation*, vol. 2, no. 2, pp. 48–61, 2021.

[7] S. Bertrand, I. Lee, B. Mishra, D. Calvert, J. Pratt, and R. Griffin, "Detecting usable planar regions for legged robot locomotion," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 4736–4742.

[8] S. Yang and S. Scherer, "Monocular object and plane SLAM in structured environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3145–3152, 2019.

[9] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[10] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[11] E. Hourdakis, S. Piperakis, and P. Trahanias, "roboSLAM: Dense RGB-D SLAM for humanoid robots," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 2224–2231.

[12] M. Camurri, M. Ramezani, S. Nobili, and M. Fallon, "Pronto: A multi-sensor state estimator for legged robots in real-world scenarios," *Frontiers in Robotics and AI*, vol. 7, p. 68, 2020.

[13] S. Yang and S. Scherer, "CubeSLAM: Monocular 3-d object SLAM," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 925–938, 2019.

[14] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger, "Elasticfusion: Real-time dense SLAM and light source estimation," *The International Journal of Robotics Research*, vol. 35, no. 14, pp. 1697–1716, 2016.

[15] T. Whelan, M. Kaess, M. Fallon, H. Johannsson, J. Leonard, and J. McDonald, "Kintinuous: Spatially extended kinectfusion," 2012.

[16] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE international symposium on mixed and augmented reality*. Ieee, 2011, pp. 127–136.

[17] M. Fourmy, D. Atchuthan, N. Mansard, J. Solà, and T. Flayols, "Absolute humanoid localization and mapping based on IMU lie group and fiducial markers," in *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, 2019, pp. 237–243.

[18] A. Roychoudhury, M. Missura, and M. Bennewitz, "3D polygonal mapping for humanoid robot navigation," in *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, 2022, pp. 171–177.

[19] A. J. B. Trevor, J. G. Rogers, and H. I. Christensen, "Planar surface SLAM with 3D and 2D sensors," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 3041–3048.

[20] P. Fankhauser, M. Bloesch, and M. Hutter, "Probabilistic terrain mapping for mobile robots with uncertain localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3019–3026, 2018.

[21] F. Jenelten, R. Grandia, F. Farshidian, and M. Hutter, "TAMOLS: Terrain-aware motion optimization for legged systems," *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3395–3413, 2022.

[22] G. Erni, J. Frey, T. Miki, M. Mattamala, and M. Hutter, "MEM: Multi-modal elevation mapping for robotics and learning," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 11 011–11 018.

[23] M. Cipriano, P. Ferrari, N. Scianca, L. Lanari, and G. Oriolo, "Humanoid motion generation in a world of stairs," *Robotics and Autonomous Systems*, vol. 168, p. 104495, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889023001343

[24] F. Jenelten, T. Miki, A. E. Vijayan, M. Bjelonic, and M. Hutter, "Perceptive locomotion in rough terrain–online foothold optimization," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5370–5376, 2020.

[25] D. Wisth, M. Camurri, and M. Fallon, "VILENS: Visual, inertial, lidar, and leg odometry for all-terrain legged robots," *IEEE Transactions on Robotics*, vol. 39, no. 1, pp. 309–326, 2023.

[26] D. Calvert, B. Mishra, S. McCrory, S. Bertrand, R. Griffin, and J. Pratt, "A fast, autonomous, bipedal walking behavior over rapid regions," in *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*. IEEE, 2022, pp. 24–31.

[27] B. Mishra, D. Calvert, B. Ortolano, M. Asselmeier, L. Fina, S. McCrory, H. E. Sevil, and R. Griffin, "Perception engine using a multi-sensor head to enable high-level humanoid robot behaviors," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 9251–9257.

[28] R. F. Salas-Moreno, B. Glocken, P. H. Kelly, and A. J. Davison, "Dense planar slam," in *2014 IEEE international symposium on mixed and augmented reality (ISMAR)*. IEEE, 2014, pp. 157–164.

[29] M. Hsiao, E. Westman, G. Zhang, and M. Kaess, "Keyframe-based dense planar SLAM," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5110–5117.

[30] D. Kanoulas, N. G. Tsagarakis, and M. Vona, "Curved patch mapping and tracking for irregular terrain modeling: Application to bipedal robot foot placement," *Robotics and Autonomous Systems*, vol. 119, pp. 13–30, 2019.

[31] A. Agarwal, A. Kumar, J. Malik, and D. Pathak, "Legged locomotion in challenging terrains using egocentric vision," in *Conference on Robot Learning*. PMLR, 2023, pp. 403–415.

[32] D. Hoeller, N. Rudin, C. Choy, A. Anandkumar, and M. Hutter, "Neural scene representation for locomotion on structured terrain," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8667–8674, 2022.

[33] B. Mishra, D. Calvert, S. Bertrand, S. McCrory, R. Griffin, and H. E. Sevil, "GPU-accelerated rapid planar region extraction for dynamic behaviors on legged robots," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 8493–8499.

[34] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 3281–3288.

[35] Boardwalk Robotics, "Nadia," URL : https://boardwalkrobotics.com/Nadia.html, visited on: September 14, 2023.

[36] R. J. Griffin, G. Wiedebach, S. McCrory, S. Bertrand, I. Lee, and J. Pratt, "Footstep planning for autonomous walking over rough terrain," in *2019 IEEE-RAS 19th International Conference on Humanoid robots (humanoids)*. IEEE, 2019, pp. 9–16.

[37] IHMC Robotics Lab Github, URL : https://github.com/ihmcrobotics/ihmc-open-robotics-software/tree/develop/ihmc-perception, visited on: Mar 6, 2024.