HELLENIC REPUBLIC
MINISTRY OF DEVELOPMENT AND INVESTMENT
GENERAL SECRETARIAT FOR RESEARCH AND INNOVATION
HELLENIC FOUNDATION FOR RESEARCH AND INNOVATION

This project is carried out within the framework of the National Recovery and Resilience Plan Greece 2.0, funded by the European Union – NextGenerationEU (Implementation body: HFRI)

Greece 2.0
Basic Research Financing Action
(Horizontal support of all Sciences)
Sub-action 1
Funding New Researchers

LEARNER

Project Title

SLAM AND PATH PLANNING MIDDLEWARE PACKAGE FOR ROBOTS IN CHALLENGING ENVIRONMENTS

Project Duration

20 November 2023 – 19 November 2025
24 Months

Project Acronym

LEARNER

Project No

015339

Deliverable No

D3.3

Deliverable Title

Journal Paper on Validated System

Deliverable Completion Date

19 November 2025

Table of Contents

Document Revision History

Version	Date	Notes
1.0	19/11/2025	First version document describing the paper submitted for the complete validated LEARNER system.

List of Acronyms

Acronym	Meaning
AlphaPose	Deep learning model for human pose estimation
HRI	Human–Robot Interaction
Nav2	Navigation 2 – ROS 2 navigation and planning framework
PP	Path Planning
SLAM	Simultaneous Localization and Mapping
ST-GCN	Spatial-Temporal Graph Convolutional Network
SuperGlue	Deep learning matcher for feature correspondences
SuperPoint	Trainable keypoint detector used in the adapted SLAM front-end
YOLO	You Only Look Once

1. Introduction

This deliverable presents the journal publication arising within Task 3.3, which reports the full validation of the LEARNER middleware package in an environment resembling real-world emergency response conditions (Task 3.2). The associated paper, included as Appendix A, provides a comprehensive technical description of the system and an extensive set of experiments demonstrating the behavior of the integrated SLAM, hybrid mapping, and socially-aware PP modules under dynamic and visually degraded conditions. The purpose of this deliverable is to summarize the validated system, highlight the scientific advances introduced in the published work, and clarify the contribution of the paper within the context of the LEARNER project.

The experiments reported in the paper evaluate the LEARNER middleware package as a complete perception-and-planning stack. The system integrates illumination-invariant SLAM, a hybrid dynamic map representation, deep-learning-enhanced perception, and a social-aware PP module capable of interpreting human actions and adjusting motion trajectories accordingly. These capabilities are validated on a robotic platform in an indoor scenario designed to reflect the challenges of emergency response environments, including power loss, smoke, dynamic obstacles, and human responders. The results demonstrate that the system maintains consistent localization, interpretable mapping, and socially compliant navigation across these conditions.

2. Overview of the Validated System

The validated system described in the paper corresponds to the complete LEARNER middleware package architecture developed across WP2 and WP3. The perception front-end is built around a modified visual-inertial SLAM pipeline derived from ORB-SLAM3, in which the classical ORB features are replaced by a trainable SuperPoint detector and a SuperGlue matcher. These learning-based components are specifically adapted to extreme illumination variations through synthetic low-light augmentation and reinforcement-learning-based feature persistence, enabling reliable localization even when the visual signal is severely degraded.

The mapping subsystem maintains a hybrid world model that blends geometric structure with semantic and temporal information. This representation is generated by combining SLAM output with a lightweight YOLO-based object detector that identifies humans, doors, chairs, tables, and other relevant elements in the robot's surroundings. By filtering keypoints belonging to dynamic or movable objects, the system preserves a stable static map for localization while simultaneously exposing semantic layers to the planner. This hybrid structure supports dynamic-object-aware trajectory adjustment and the generation of socially meaningful navigation decisions.

On top of these perception capabilities, the system integrates a social-aware PP module based on the Nav2 framework. Human presence, pose, and action are estimated through an AlphaPose-based pipeline followed by temporal classification using ST-GCN. The resulting action labels modulate the spatial cost structure of the navigation space, informing the planner when humans are stationary, walking, running, interacting with nearby objects, or performing high-urgency movements. This allows the robot to adjust safety margins, anticipate human motion, and adopt behavior that aligns with human comfort and operational constraints.

3. Experimental Validation

The validation campaign described in the paper assesses the full LEARNER middleware under controlled, progressively more challenging experimental stages. The experiments begin in a standard well-lit environment and gradually introduce obstacles, human actors, and dynamic motion patterns. The system is subsequently tested again under a replicated scenario in which the illumination was extremely low.

The results confirm that the deep-feature-augmented SLAM pipeline maintains stable localization in both normal and dark conditions, outperforming classical ORB-based pipelines that fail entirely when illumination drops. The hybrid mapping subsystem reliably reconstructs the environment while correctly excluding transient or movable objects, leading to consistent point clouds that align with ground-truth layouts even in visual conditions where conventional SLAM would collapse.

Object detection remains functional across lighting conditions, with reduced—but still meaningful—performance in the dark. Human action recognition sustains strong accuracy. When integrated into

navigation, these perception outputs enable the robot to avoid dynamic objects and humans, maintain appropriate interpersonal distance, and follow a selected human target with robust recovery behavior after temporary occlusions.

Overall, the validation demonstrates that the LEARNER system can sustain closed-loop autonomous operation in environments combining dynamic human activity, degraded visual conditions, and structural changes. These results verify the scientific contributions of WP2 and WP3 and confirm the readiness of the middleware package for deployment within realistic operational settings.

4. Conclusion

This deliverable documents the journal publication that presents the final, experimentally validated version of the LEARNER middleware package. The work confirms that the system achieves robust SLAM in visually challenging conditions, constructs a hybrid dynamic map capable of semantic reasoning, and performs socially compliant navigation based on human pose and action interpretation. The full publication, containing methodology, experimental details, and quantitative analyses, is included in Annex A.

Annex A

Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

ORIGINAL RESEARCH PAPER

LEARNER: A SLAM and Path Planning Middleware Package for Dynamic and Visually Challenging Environments

Panagiotis Bakirtzis[1] | **Anastasios Agakides**[2] | **Loukas Bampis**[1]

[1]Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi, Greece

[2]Department of Production and Management Engineering, Democritus University of Thrace, Xanthi, Greece

Correspondence
Corresponding authors: Panagiotis Bakirtzis
Email: pbakirtz@ee.duth.gr and Anastasios Agakides
Email: aagakidi@pme.duth.gr.
Both authors contributed equally in this work.

Abstract

Autonomous robotic platforms are utilized in increasingly diverse aspects of modern life, from domestic service robots to self-driving vehicles, continuously assuming laborious, repetitive, or tasks of increased risk for human actors. Despite these advancements, several mission-critical applications remain constrained by the inability of current systems to robustly operate under highly dynamic and unpredictable conditions. Aiming to reduce these limitations, this paper presents LEARNER, a middleware package fully compatible with the Robot Operating System (ROS 2), aiming to enhance the perception and navigation capabilities of mobile robots in dynamic environments populated by humans. LEARNER introduces three main pipelines: i) a trainable perception architecture for mapping previously unknown environments under extreme environmental variations, ii) a hybrid map representation that preserves the dynamic and semantic attributes of observed entities, and iii) a socially-aware path planning framework that enables human-centric navigation. The proposed middleware was validated on a Unitree GO2 EDU robotic platform over a real-world experimental setup, simulating the operational conditions of an indoor fire emergency, characterized by power failure, moving obstacles, and active personnel. The obtained results demonstrate our system's capacity to sustain reliable localization, mapping, and socially compliant navigation in complex and visually degraded environments, thereby advancing the state of the art in autonomous robot operation under extreme conditions.

KEYWORDS
SLAM, Path Planning, Map Representation, Action Recognition, Deep Learning

1 | INTRODUCTION

Autonomous robotic systems are rapidly becoming an integral component of modern infrastructure, assuming a wide spectrum of applications that range from domestic services to industrial logistics, precision agriculture, and emergency response [1–3]. Major recent advancements in sensing, control, and Deep Learning (DL) have enabled remarkable progress in robotic autonomy, particularly in Simultaneous Localization and Mapping (SLAM) and Path Planning (PP) systems. However, reliable robot operation in highly dynamic, human-populated, and visually degraded environments still remains an open challenge.

In such scenarios, classical SLAM and PP pipelines often struggle to maintain consistent performance [4], since the majority of existing systems assume a semi-static environment with stable illumination and minimal structural changes. Therefore, they are highly sensitive to conditional changes, such as lighting variation, occlusions, as well as the dynamic nature of human behavior. Moreover, current DL-based SLAM and PP approaches typically depend on extensive labeled datasets, which limits their generalization in conditions not encountered during training.

In order to overcome those challenges, this work presents LEARNER, a SLAM and PP middleware compatible with Robot Operating System (ROS 2), aiming towards reliable robot operation in dynamic and human-centered settings. LEARNER attempts to bridge the gap between traditional geometric methods and contemporary AI-driven perception by utilizing a hybrid model-based and learning-based architecture. Three fundamental elements form the basis of its design. To enable the robot to perceive and map unknown environments in extremely low-light conditions, we first employ a trainable perception module that improves illumination-invariance and local features optimization. Next, a hybrid map representation engine is implemented, capable of capturing the

Abbreviations: SLAM, Simultaneous Localization and Mapping; PP, Path Planning; DL, Deep Learning.

static and dynamic properties of environmental entities by combining temporal, semantic, and metric information into a single structure. Finally, a social-aware path planning framework is introduced, which integrates human pose and action recognition into the navigation stack, in order to enable context-adaptive trajectory generation that respects human proximity, movement, and behavioral cues. The system is validated through a physical experiment replicating the conditions of an indoor fire emergency, featuring light failure, moving obstacles, and active human responders. Our findings demonstrate the middleware's ability to sustain consistent mapping accuracy, semantic awareness, and socially compliant navigation even under severe perceptual degradation. The main contributions of this work can be summarized as follows:

- A visual-inertial SLAM framework based on the detection of robust DL-augmented local features, that improves robustness to illumination variability through photo-realistic synthetic training and feature persistence.
- A hybrid map model integrating metric, semantic, and temporal layers to maintain dynamic situational awareness.
- A socially adaptive PP engine incorporating low-light human-action recognition for safe and intuitive robot navigation in human-populated environments.
- A ROS 2 compatible middleware implementation, validated through a real-world emergency-like scenario on the Unitree GO2 EDU robotic platform.

The rest of this paper is organized as follows. Section 2 introduces some of the most representative related literature in the field of robot operation in uncontrolled environments operated by humans. Section 3 details our system's components for achieving SLAM in challenging conditions, retaining a hybrid map that excludes dynamic objects during localization, and performing socially-aware PP by recognizing human actions during challenging environmental conditions. In Section 4, the integration steps for incorporating the developed sub-systems into a single robotic platform are detailed, while Section 5 presents our experimental setup and the obtained results. Finally, Section 6 concludes our work and explores possible avenues for future work.

2 | RELATED WORK

Robust robot autonomy in human-populated, dynamic, and visually degraded environments depends on three coupled capacities: i) maintaining localization and mapping under severe appearance changes and dynamic elements, ii) representing the scene in a way that fuses geometry, semantics, and temporal evolution, and iii) navigating while respecting human safety, comfort, and intent. This section reviews prior work in these areas.

2.1 | Dynamic SLAM and Illumination Invariance

Classical SLAM pipelines assume that the environment is locally static and visually well-behaved. In feature-based visual and visual-inertial SLAM [5], pose estimation is achieved by tracking salient local feature points across frames and solving for the camera motion using a combination of multi-view geometry and estimation theory. These tracked correspondences are also used to triangulate the scene's structure and incrementally build a map.

In the typical case, this tracking is driven by hand-crafted detectors and descriptors, such as SIFT [6], SURF [7], and ORB [8]. SIFT provides scale- and rotation-invariant features but is computationally heavy, while SURF and related accelerations maintain robustness while improving efficiency. ORB couples the FAST [9] corner detector with a rotated BRIEF [10] descriptor, achieving lightweight feature extraction, which is suitable for embedded systems. These families of methods have been deeply integrated into state-of-the-art SLAM systems, such as ORB-SLAM3 [5], VINS-Mono [11], or CD-SLAM [12], which achieve highly accurate localization, loop closure, and relocalization in sufficiently textured and structured environments.

However, the quality of these pipelines degrades in scenes that violate their assumptions. Two modes of failure are repeatedly reported in the literature. Firstly, appearance instability can alter local gradients and suppress repeatability; thus, significantly reducing the number of tracked keypoints [13]. Furthermore, scene dynamics, as in the cases of moving humans, vehicles, tools, and deformable objects (e.g., doors opening), introduce non-stationary structures that invalidate static-world assumptions and contaminate the map [4]. To mitigate these issues, modern approaches increasingly leverage learning-based perception in the SLAM front-end. Deep local feature extractors, such as SuperPoint [14], D2-Net [15], LF-Net [16], R2D2 [17], ASLFeat [18], and related architectures, learn both keypoint locations and descriptors rather than relying on fixed gradient heuristics. These models are typically trained with self-supervision or weak supervision, sharing the potential of producing local features that remain repeatable across viewpoint and lighting variation, as well as discriminative under appearance alternations.

2.2 | Hybrid Map Modeling

Mapping for navigation can be broadly categorized into four canonical representations: metric, topological, semantic, and hybrid [19–24]. Metric maps (e.g., occupancy grids and point clouds) are the most common category since they provide high-fidelity geometric reconstruction of free space and obstacles, allowing for accurate localization and collision avoidance in mainly static settings [19]. Moreover, topological maps abstract the world as a graph of places (nodes) and traversable relations (edges), approximating how humans and animals form internal

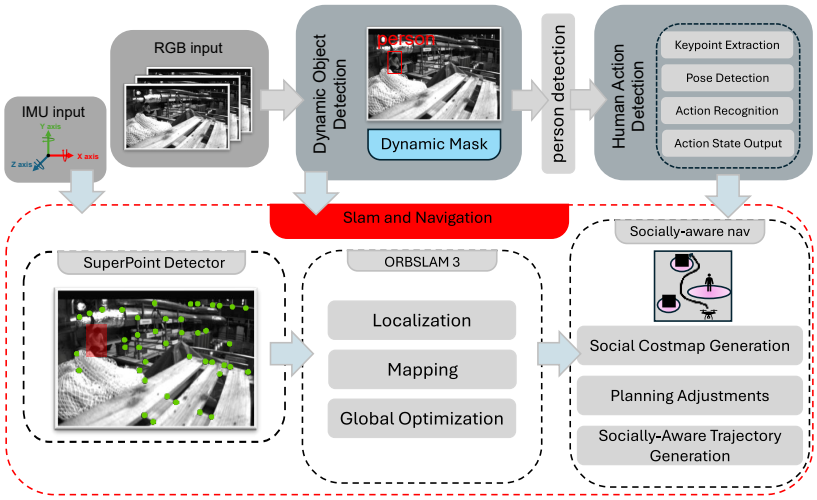

FIGURE 1 Overview of the LEARNER middleware architecture, illustrating the integrated SLAM and PP pipeline. The system combines dynamic object detection, low-light human action recognition, illumination-invariant feature extraction, hybrid semantic mapping, and socially-aware path planning to enable robust autonomous navigation in dynamic and visually degraded environments.

cognitive maps. These maps are compact and resilient to local geometric drift, but lack spatial precision, rendering them most suitable for SLAM applications [20]. Semantic maps attach human-interpretable labels (e.g., door, person, chair) to regions or objects in the environment using object detection, semantic segmentation, or scene understanding networks [21, 22]. Using this information, the robotic agent is not restricted to reasoning only for obstacles, but also for affordances, task-relevant zones, and social constraints. Finally, hybrid maps attempt to unify these layers, associating local metric sub-maps with a global topological backbone and enriching them with semantic attributes [23, 24]. Such representations maintain geometric accuracy at the local scale, while enabling reasoning over structure and function, allowing for the distinction between static entities and objects that may be relocated over time.

Recent work in hybrid mapping enhances classical approaches for measuring the environment's geometry by combining deep detectors and segmenters, which can identify humans, tools, doors, and other salient entities in the scene [4]. This allows downstream planners to reason over labeled obstacles that may induce specific properties, such as doors that can open or close, furniture that may move, or humans that perform specific movements based on the action they perform.

2.3 | Socially-Aware Path Planning and Action Recognition

Classical PP aims to find a collision-free trajectory from an initial pose to a goal pose. Early techniques include grid-based search and potential field methods in which obstacles apply repulsive forces while the goal applies an attractive force, and the robot follows the resultant force field [25]. While simple and computationally efficient, these methods treat all obstacles uniformly and assume that safety can be modeled purely as geometric clearance.

However, in human-populated environments, this assumption can no longer be considered. Human-Robot Collaboration (HRC) imposes additional constraints: maintaining interpersonal distance, avoiding sudden intrusions into human workspaces, and adapting behavior to urgency or stress levels [26]. The planning policy is therefore required to be not just collision-free, but socially compliant. Therefore, the current frontier in navigation research is to fuse these perception outputs directly into the motion planner. Instead of treating a person as just a moving obstacle with a circular safety radius, planners incorporate human proximity preferences and personal space, predicted motion direction and speed, and activity class (e.g., walking casually vs. running in urgency). This level of information allows for navigation around socially-aware costmaps, where each cell in the planner's world model is assigned not only with a geometric occupancy, but also with a social cost derived from human state, predicted trajectory, and task role [27].

A key enabler of socially compliant navigation is reliable human perception. Human pose estimation has advanced significantly with high-resolution convolutional backbones, such as HRNet [28] and its lighter successors (Lite-HRNet [29], Dite-HRNet [30]), which maintain spatial detail throughout the network to produce accurate multi-joint skeletons for each person in the scene. Furthermore, top-down approaches, like the AlphaPose [31], first localize each human and then estimate a full-body pose with joint-level precision.

Once poses are available, action recognition provides higher-level intent cues. Skeleton-based action recognition methods, such as Spatial-Temporal Graph Convolutional Networks (ST-GCN) [32], model the human body as a graph, where nodes represent joints, edges encode anatomical and temporal connectivity, and learning is responsible for evaluating how joint trajectories evolve over time. Video-based architectures, such as the SlowFast network [33], process both slow semantic

context and fast motion cues in parallel to classify complex activities. Finally, toolkits such as MMAction2[‡] unify these and other state-of-the-art vision backbones for robust activity understanding in video streams. However, even though there exist some datasets available in the literature for training an action recognition module over visually challenging environmental conditions, there exist few-to-no approaches that can effectively address such a task.

3 | METHODOLOGY

In this section, our methodology for developing the complete LEARNER pipeline is detailed. Figure 1 presents the main components of the proposed architecture, including a trainable local feature detection approach, a hybrid map representation that identifies and excludes dynamic objects and humans from the main environment's representation, and an action recognition module for supporting socially aware PP.

3.1 | Trainable Local Feature Detection for Visually Challenging SLAM Scenarios

To address the lighting variations, we adapted a learning-based feature detector that maintains robustness across diverse illumination conditions. Building upon our previous work on illumination-invariant keypoint detection [34], we employ SuperPoint [14] as the foundation for feature extraction. The original SuperPoint network was enhanced through a domain adaptation method, trained on synthetically augmented datasets that simulate extreme lighting variations, ranging from complete darkness to overexposure. This training procedure, detailed in [34], incorporates an Illumination Conditions Adaptation (ICA) mechanism, which makes use of local features extracted from those synthetic data and learns to associate them among different conditions. This process enables the detector to learn illumination-invariant feature representations, while also enhancing feature matching.

3.2 | Hybrid Mapping for Dynamic Object Handling

Traditional SLAM systems assume a static environment, causing dynamic objects, such as moving people or relocatable furniture, to contaminate the map and degrade localization accuracy. To address this, we implemented a hybrid mapping approach that combines geometric reconstruction with semantic object detection to distinguish between static and dynamic scene elements.

We make use of YOLOv11s [35] for indoor object detection, focusing on four classes relevant to human-populated environments: *Person*, *Door*, *Table*, and *Chair*. Transfer learning is also employed to fine-tune the pre-trained model on the custom dataset. The training incorporated a degrading learning rate schedule and backbone freezing to preserve pre-learned features, while adapting to the target domain. YOLOv11s is selected to balance detection accuracy with real-time inference performance on embedded platforms. During mapping, detected bounding boxes corresponding to dynamic or semi-static objects are used to filter feature within those regions. Specifically, keypoints detected by the enhanced SuperPoint module that fall within the bounding boxes of identified dynamic objects are excluded from map construction and pose estimation. This prevents transient or movable entities from being incorporated into the persistent map representation, while allowing the SLAM system to track stable structural elements, such as walls, floors, and fixed infrastructure. The resulting hybrid map maintains both geometric accuracy for localization and semantic awareness for downstream navigation tasks, as the detected object classes and their spatial locations are preserved in a semantic layer accessible to the PP module.

3.3 | Action Recognition for Socially Aware Navigation

In order for the robot to effectively operate among human personnel and interact with them, an action recognition pipeline is proposed, capable of identifying humans through an RGB sensor in low-lighting environmental conditions. To achieve this, we utilize AlphaPose [31], a well-established, lightweight, and highly accurate network for human pose estimation that operates over the detections of YOLOv11s for the *Person* class. This network is then trained on an augmented learning set containing the original samples of COCO [36] dataset, as well as an artificially darkened copy obtained through CycleGAN-Turbo [37].

The pose estimation results are then filtered based on their confidence scores to discard joint detections of low certainty. Specifically, the Otsu's method is deployed over a distribution of the confidence scores obtained from 3M joints detected on 2.5k videos of COCO. This approach resulted in a detection certainty threshold $th_j = 0.507$, which can be used to distinguish the joints that are sufficiently accurate to propagate over the action recognition module.

Therefore, during the robot's online deployment, the subset of filtered human pose estimation results is fed to the action recognition network of ST-GCN, which is included in the MMAction2 toolkit. Note that since this network relies only on the coordinates of the detected joints, it is not influenced by any alterations in the lighting conditions that the environment may induce. Finally, the obtained action recognition results are used to generate additional socially-derived proximity zones on the mapped world's occupancy grid using [38].

Based on the above, the robotic agent can safely avoid human actors when they are found on its path using the occupancy grid, or discreetly follow them when the robot needs to cooperate with the responders. The human tracking-following operation is carried out using the pose estimation results to estimate the distance between the robot and the human, maintaining a predefined following distance. If the robot loses sight of its target, it employs a recovery strategy by rotating in place, starting from its last known direction. This rotational movement allows

[‡] https://github.com/open-mmlab/mmaction2

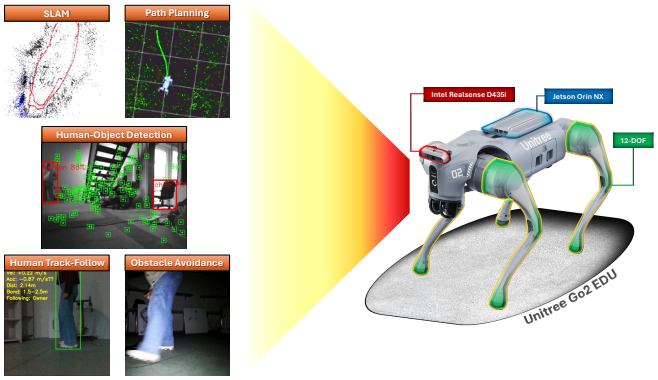

FIGURE 2 Robotic platform used within our experiments: Unitree Go2 EDU quadruped robot equipped with an Intel RealSense D435i camera and onboard NVIDIA Jetson Orin NX. The system autonomously executes SLAM, global/local PP, obstacle avoidance, human-object detection, and track-follow tasks in real time using ROS 2 (Humble), all processed on-board without external computations.

the robot to search for the human in a systematic manner, continuously scanning its surroundings until the human is re-detected. The robot then resumed following the human once visual confirmation was established.

4 | INTEGRATION

4.1 | Middleware Package

The complete SLAM pipeline integrates the components described in Section 3, into a unified system. Starting from the ORB-SLAM3 framework [5], we replaced its front-end feature extraction and matching modules with our learning-based keypoints, while preserving its robust back-end optimization, tracking, and loop closure mechanisms. Specifically, the ORB feature detector was removed and substituted with our enhanced SuperPoint detector described in Section 3.1, providing illumination-invariant keypoint localization. Following feature detection, the YOLO-based object detector outlined in Section 3.2 identifies dynamic and semi-static objects within each frame. Keypoints falling inside the bounding boxes of detected dynamic entities (*Person*, *Door*, *Table*, and *Chair*) are filtered out prior to map construction and pose estimation, ensuring that only features corresponding to static structures contribute to localization. All DL-based components were converted from their original Python implementations into optimized C++ modules using ONNX models and TensorRT for efficient inference on embedded hardware. These modules were then integrated into a single executable that maintains compatibility with the ORB-SLAM3 architecture. To facilitate deployment on robotic platforms, we developed an ROS 2 wrapper to handle the entire SLAM pipeline. This wrapper subscribes to camera image streams and IMU data from the robot's onboard sensors, processes them through the integrated perception and mapping modules, and publishes localization estimates, map updates, and semantic object detections to the ROS ecosystem.

4.2 | Robotic Platfrom

To support the methodology outlined in Section 4.1, it was necessary to employ a robotic system capable of reliably executing the required tasks under dynamic real-world conditions. To this end, the Unitree Go2 EDU quadruped platform (Fig. 2), augmented with an Intel RealSense D435i camera and powered by the onboard NVIDIA Jetson Orin NX computing module, was integrated into the testbed. Its 12-DOF legged architecture enables dynamic traversal of complex unstructured terrain including stairs, slopes, and debris while maintaining stable sensor orientationty. This mobility profile introduces an additional challenge over the proposed perception-driven framework, including aggressive body motions and non-planar ground contact. The integrated Jetson Orin NX (up to 100 TOPS) provides sufficient on-robot compute to run the entire processing stack of our system, without requiring off-board transmission, enabling closed-loop autonomy even in communication-denied settings. The D435i provides configurable and synchronized visual data streams, enabling adaptive resolution and frame-rate selection to balance accuracy, field coverage, and onboard computational constraints across varying operational scenarios. Its on-board 6-DOF IMU (200 Hz accelerometer and gyroscope, configurable at 100/200/400 Hz) enables tight visual-inertial fusion, effectively mitigating gait-induced vibrations and rapid body motions while maintaining drift rates below 0.3% per meter in dynamic environments.

The custom SLAM pipeline, implemented natively on the Jetson Orin NX, leverages ROS 2 (Humble) as the central middleware to orchestrate autonomous execution of all tasks: SLAM, map representation, and action recognition, effectively enabling PP, obstacle avoidance, and human track-follow. All sensors are hardware-synchronized via a Jetson Orin NX GPIO trigger and timestamped in ROS 2. In order to monitor and analyze the data, a wireless communication with the Unitree Go2 EDU robot is achieved using a modified version of the open-source

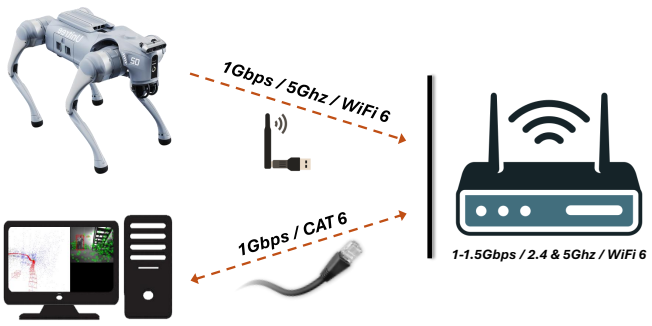

FIGURE 3 Schematic of the local network with robot, router, and desktop endpoints.

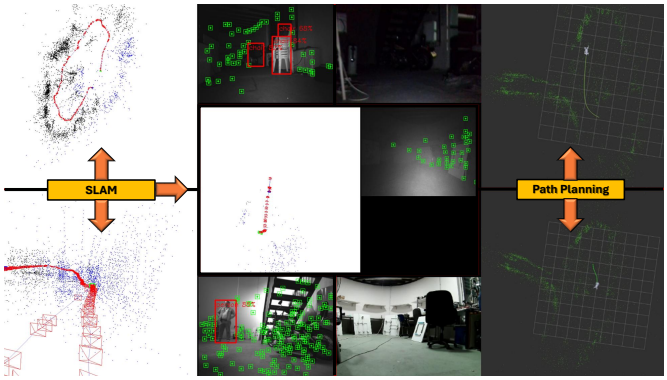

FIGURE 4 An overview of our experimental setup used within the validation of the proposed SLAM and PP pipeline.

ROS 2 SDK[§]. This SDK enables real-time bidirectional data exchange over Wi-Fi via WebRTC and Ethernet via CycloneDDS. Our modifications extend the base implementation with custom message filtering, precise timestamp alignment, and seamless integration into the autonomous navigation stack. Finally, as depicted if Fig. 3, data transfer between the computer station and the robotic system was achieved through a Wi-Fi network operating on the 5GHz frequency band with a bandwidth capacity of 1Gbps and complemented by a data adapter, with equivalent specifications, on the robot platform.

5 | RESULTS

All experimental tests were conducted within the confines of our laboratory's designated testing facility. The experimental protocol was organized into four sequential stages, as detailed below.

- Stage 1: Initial SLAM experiments were conducted under optimal illumination conditions in an obstacle-free environment.
- Stage 2: In the subsequent phase, obstacles (humans, chairs) were introduced into the environment, and the robotic system performed PP with obstacle avoidance and in accordance to the social constrains implied from the recognized humans' action.
- Stage 3: In the third phase, the system performed human tracking-following.
- Stage 4: In the final stage, we meticulously repeated the experimental procedures encompassing stages 1 through 3, replicating the entire sequence under dark illumination conditions.

Figure 4 contains an overview of our experimental setup on the designated site.

5.1 | SLAM

To assess the robustness of the proposed deep feature integration, monocular-inertial SLAM experiments were conducted using the modified ORB-SLAM3 framework equipped with deep learning-based feature extractors. These tests leveraged the system's tightly-coupled visual-inertial pipeline, enabling simultaneous localization and mapping under challenging conditions such as rapid motion, low texture, and dynamic lighting. The inertial measurements from the IMU were fused with visual constraints derived from the deep feature correspondences to initialize and maintain scale-aware tracking throughout the sequences.

The modified version was tested in both adequate light conditions ("Light Environment") and when the lights were switched off ("Dark Environment"). The mapping outcome of the above experiment is presented in Fig. 5, overlayed on-top of the area's floorplans. As it can be seen, our trainable architecture successfully operated under both settings. Compared to the "Light Environment", the results produced in the dark conditions showed a notably reduced number of mapped 3D points and an increased dispersion, especially towards the map's boundaries. Nevertheless, they were more than suitable for producing an occupancy grid and supporting autonomous navigation. Finally, with the aim to compare our approach with the baseline implementation, we additionally evaluated the standard ORB-SLAM3 system on the "Dark Environment", which however, failed to track or map any meaningful part of the environment.

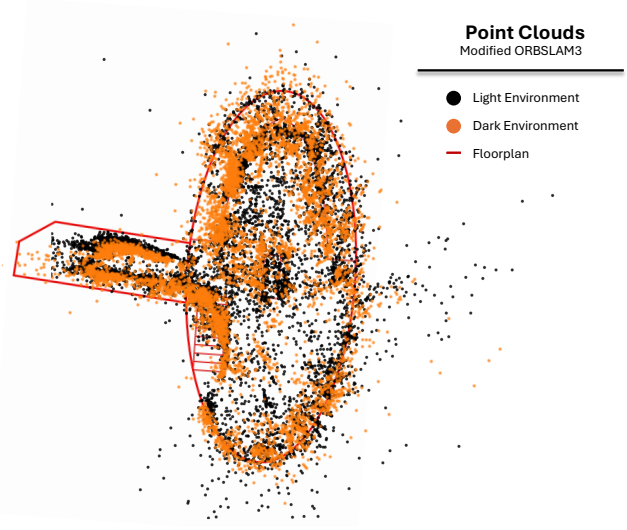

FIGURE 5 Mapping results of the proposed SLAM architecture. The point cloud produced in adequate light conditions is depicted in black, the one captured in dark conditions in orange, while the area's hand-measured floorplans are presented in red.

§ https://github.com/abizovnuralem/go2_ros2_sdk

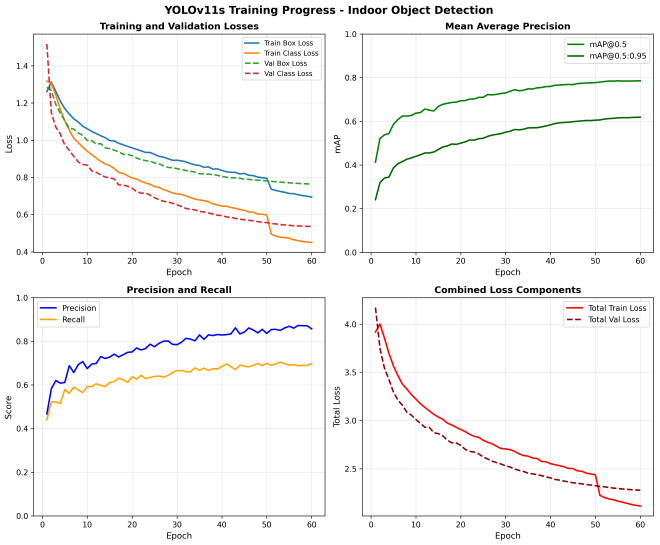

FIGURE 6 YOLOv11s training progression over 60 epochs showing (upper-left) training and validation losses, (upper-right) mean Average Precision metrics, (bottom-left) precision and recall evolution, and (bottom-right) combined loss convergence.

5.2 | Dynamic Object Identification

To evaluate the performance of our dynamic object detection module, we assessed our retrained YOLOv11s detector described in Section 3.2 in indoor scenes containing the four target classes: *Person*, *Door*, *Table*, and *Chair*. Figure 6 presents the training progression over 60 epochs. The model exhibited smooth convergence with consistent alignment between training and validation losses, indicating robust generalization. The mean Average Precision at IoU threshold 0.5 (mAP@0.5) reached 78.56%, while mAP@0.5:0.95 achieved 61.89%. The final precision and recall values of 85.67% and 69.72% reflect the model's ability to minimize false positives while maintaining adequate detection coverage.

Figure 7 depicts representative examples dynamic objects detected during our "Light Environment" and "Dark Environment" experiments. In both cases, the identified objects were omitted from the final map since they are considered either dynamic or not part of the permanent environment structure. This filtering step allows the robot to localize itself using only static elements of the surroundings. However, for the purpouse of PP, these objects were considered for dynamic obstacle avoidance.

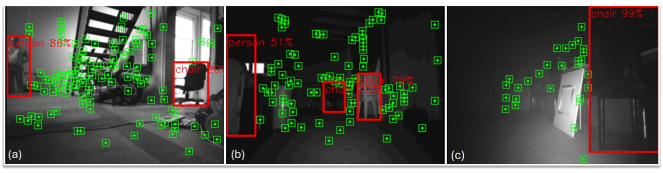

FIGURE 7 Illustration of object detection under different environmental conditions (from left to right): (a) "Light Environment" and (b-c) "Dark Environment" experiments.

FIGURE 8 Representative example of action recognition in dark conditions using the proposed approach.

5.3 | Action Recognition

Within the scope of this work, we utilized 5 distinct classes that were included in the pretrained ST-GCN of the MMAction2 and were relevant to our emergency response experimental setup, namely: *Jump Up*, *Pick Up*, *Sit Down*, *Stand Up*, and *Running*. The obtained recognition accuracy results were reported at 91.57% in adequately light scenes and at 84.57% in dark conditions. Figure 8 contains a representative instance, where a human actor is recognized using the proposed approach under severe low-lighting conditions.

The human-following system performed reliably across indoor environments, both in well-lit and low-light conditions. It consistently tracked the target throughout most trials, maintaining a stable following distance within the 1.5-2.5 m range, in accordance with the selected proximity constraints. When the human was temporarily lost due to occlusion or movement, the robot effectively re-acquired the target by rotating in place from the last known direction. This systematic recovery minimized interruptions and enabled smooth resumption of tracking. Figure 9 contains instances of our human-following module under different indoor lighting conditions, reliably handling occlusions and dynamic motion, while maintaining stable distance control. However,

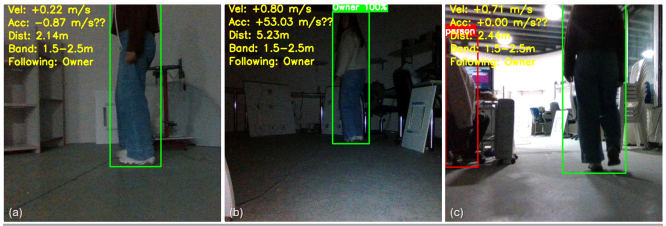

FIGURE 9 From left to right: (a) human tracking under clear illumination conditions, (b) human tracking in dark conditions, and (c) human tracking when multiple humans are present in the frame.

a slight reduction in detection confidence was observed in the "Dark Environment" experiments, in addition to occasional delays in pose re-acquisition during prolonged full occlusions of the targeted human actor.

6 | CONCLUSIONS

In this paper, we present an integrated system for SLAM and path planning (PP) operating under challenging environmental conditions. The proposed architecture incorporates i) a trainable local feature–extraction module aimed at enhancing SLAM robustness under varying illumination, ii) a hybrid mapping representation that couples spatial and semantic information, and iii) an action-recognition component that introduces socially derived proximity constraints to support human-aware navigation. Dynamic entities encoded in the hybrid map are excluded from the localization process and are instead treated solely as obstacles during PP. Experiments were performed in a controlled environment designed to emulate emergency-response conditions, using the Unitree GO2 EDU quadruped platform. The obtained results demonstrate the system's ability to maintain reliable navigation performance despite large environmental variations and the dynamic behavior of human participants. Future work will focus on evaluating the approach in a multi-robot setting and extending the robot's behavioral repertoire for traversal of dynamic and potentially damaged environments, including navigation over debris.

ACKNOWLEDGMENTS

This research is implemented in the framework of H.F.R.I call "Basic research Financing (Horizontal support of all Sciences)" under the National Recovery and Resilience Plan "Greece 2.0" funded by the European Union – NextGenerationEU (H.F.R.I. Project Number: 15339).

CONFLICT OF INTEREST

The authors declare no potential conflict of interests.

REFERENCES

1. Haopeng Zhao, Zhichao Ma, Lipeng Liu, Yang Wang, Zheyu Zhang, Hao Liu, in *Proc. IEEE International Conference on Advanced Algorithms and Control Engineering*, **2025**, pp. 464–467.
2. Kshetrimayum Lochan, Asim Khan, Islam Elsayed, Bhivraj Suthar, Lakmal Seneviratne, Irfan Hussain, *IEEE Access* **2024**.
3. Christian A Schroth, Christian Eckrich, Ibrahim Kakouche, Stefan Fabian, Oskar Von Stryk, Abdelhak M Zoubir, Michael Muma, *IEEE Transactions on Biomedical Engineering* **2024**, *71* (6), 1756–1769.
4. Yanan Wang, Yaobin Tian, Jiawei Chen, Kun Xu, Xilun Ding, *IEEE Transactions on Instrumentation and Measurement* **2024**, *73*, 1–21.
5. Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, Juan D Tardós, *IEEE Transactions on Robotics* **2021**, *37* (6), 1874–1890.
6. David G Lowe, *International Journal of Computer Vision* **2004**, *60* (2), 91–110.
7. Herbert Bay, Tinne Tuytelaars, Luc Van Gool, in *Proc. European conference on computer vision*, **2006**, pp. 404–417.
8. Ethan Rublee, Vincent Rabaud, Kurt Konolige, Gary Bradski, in *Proc. IEEE International Conference on Computer Vision*, **2011**, pp. 2564–2571.
9. Edward Rosten, Tom Drummond, in *Proc. European Conference on Computer Vision*, **2006**, pp. 430–443.
10. Michael Calonder, Vincent Lepetit, Christoph Strecha, Pascal Fua, in *Proc. European Conference on Computer Vision*, **2010**, pp. 778–792.
11. Tong Qin, Peiliang Li, Shaojie Shen, *IEEE Transactions on Robotics* **2018**, *34* (4), 1004–1020.
12. Shuhuan Wen, Sheng Tao, Xin Liu, Artur Babiarz, F Richard Yu, *IEEE Transactions on Instrumentation and Measurement* **2024**, *73*, 1–8.
13. Anastasios Agakidis, Panagiotis Bakirtzis, Loukas Bampis, in *Proc. IEEE European Conference on Mobile Robots*, **2025**, pp. 1–6.
14. Daniel DeTone, Tomasz Malisiewicz, Andrew Rabinovich, in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, **2018**, pp. 224–236.
15. Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, Torsten Sattler, in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, **2019**, pp. 8092–8101.
16. Yuki Ono, Eduard Trulls, Pascal Fua, Kwang Moo Yi, *Advances in Neural Information Processing Systems* **2018**, *31*.
17. Jerome Revaud, Cesar De Souza, Martin Humenberger, Philippe Weinzaepfel, *Advances in Neural Information Processing Systems* **2019**, *32*.
18. Zixin Luo, Lei Zhou, Xuyang Bai, Hongkai Chen, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, Long Quan, in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, **2020**, pp. 6589–6598.
19. Kenji Koide, Masashi Yokozuka, Shuji Oishi, Atsuhiko Banno, *Robotics and Autonomous Systems* **2024**, *179*, 104750.
20. Howie Choset, Keiji Nagatani, *IEEE Transactions on Robotics and Automation* **2001**, *17* (2), 125–137.
21. Jose Cuaran, Kulbir Singh Ahluwalia, Kendall Koe, Naveen Kumar Uppalapati, Girish Chowdhary, in *Proc. IEEE International Conference on Robotics and Automation*, **2025**, pp. 12716–12722.
22. Siting Zhu, Guangming Wang, Hermann Blum, Jiuming Liu, Liang Song, Marc Pollefeys, Hesheng Wang, in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, **2024**, pp. 21167–21177.
23. Romain Drouilly, Patrick Rives, Benoit Morisset, in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, **2015**, pp. 5109–5114.
24. Jianhao Jiao, Ruoyu Geng, Yuanhang Li, Ren Xin, Bowen Yang, Jin Wu, Lujia Wang, Ming Liu, Rui Fan, Dimitrios Kanoulas, *IEEE Transactions on Automation Science and Engineering* **2024**.
25. Minghan Wei, Daewon Lee, Volkan Isler, Daniel Lee, in *Proc. IEEE International Conference on Robotics and Automation*, **2021**, pp. 8551–8557.
26. Anthony Francis, Claudia Pérez-d'Arpino, Chengshu Li, Fei Xia, Alexandre Alahi, Rachid Alami, Aniket Bera, Abhijat Biswas, Joydeep Biswas, Rohan Chandra, et al., *Transactions on Human-Robot Interaction* **2025**, *14* (2), 1–65.
27. Phani Teja Singamaneni, Pilar Bachiller-Burgos, Luis J Manso, Anaís Garrell, Alberto Sanfeliu, Anne Spalanzani, Rachid Alami, *The International Journal of Robotics Research* **2024**, *43* (10), 1533–1572.
28. Ke Sun, Bin Xiao, Dong Liu, Jingdong Wang, in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, **2019**, pp. 5693–5703.
29. Changqian Yu, Bin Xiao, Changxin Gao, Lu Yuan, Lei Zhang, Nong Sang, Jingdong Wang, in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, **2021**, pp. 10440–10450.

30. Qun Li, Ziyi Zhang, Fu Xiao, Feng Zhang, Bir Bhanu, in *Proc. International Joint Conference on Artificial Intelligence*, **2022**, pp. 1095–1101.
31. Hao-Shu Fang, Jiefeng Li, Hongyang Tang, Chao Xu, Haoyi Zhu, Yuliang Xiu, Yong-Lu Li, Cewu Lu, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2022**, *45* (6), 7157–7173.
32. Sijie Yan, Yuanjun Xiong, Dahua Lin, in *Proc. Conference on Artificial Intelligence*, **2018**.
33. Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, Kaiming He, in *Proc. IEEE/CVF International Conference on Computer Vision*, **2019**, pp. 6202–6211.
34. Anastasios Agakidis, Loukas Bampis, Antonios Gasteratos, in *Proc. IEEE International Conference on Imaging Systems and Techniques*, **2023**, pp. 1–6.
35. Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, **2016**, pp. 779–788.
36. Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, C Lawrence Zitnick, in *Proc. European Conference on Computer Vision*, **2014**, pp. 740–755.
37. Gaurav Parmar, Taesung Park, Srinivasa Narasimhan, Jun-Yan Zhu, *arXiv preprint arXiv:2403.12036* **2024**.
38. Jonatan Ginés Clavero, Francisco Martín Rico, Francisco J Rodríguez-Lera, José Miguel Guerrero Hernández, Vicente Matellán Olivera, in *Workshop of Physical Agents*, **2020**, pp. 3–17.