2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

# WATCHDOG: Autonomous Elderly Assistance via Attention-based Fall Detection and Trajectory Prediction

A. Longo[1,3], A. Bono [1 2], G. Guaragnella [2], P. Boccadoro [1], A. Rana [2], A. Petitti [2], T. D'Orazio [2]

***Abstract*— Service robots designed to assist elderly people are receiving significant attention since they can improve their quality of life, promote their independence, and provide daily support. These mobile platforms can observe people moving around their homes, recognize dangerous events, and detect them promptly. This paper introduces a novel framework to perform fall detection and people following on board an autonomous legged robotic platform. The system operates on the Unitree Go2 robot and comprises two main building blocks. The first component consists of a Body Landmarks extractor and a Transformer-based network that performs binary classification, distinguishing between Fall behaviours and Activities of Daily Living (ADL). The second component is a target-driven path planner that enables the robot to follow and maintain a full-body view of the target in complex environments. Experiments on public datasets and comparison with state-of-the-art works have been conducted to demonstrate the reliability of both blocks. Real experiments in a cluttered environment have been performed to illustrate how the mobile platform is able to follow people moving around obstacles, detect falls in occluded areas, and predict people's trajectories to maintain a full-body view.**

**Code and additional material are available at the following link: `https://github.com/Antus8/WatchDog`.**

(a) (b)

Fig. 1. A legged robot for elderly monitoring via fall detection and motion prediction in challenging scenarios. In (a), a fall event occurs behind objects occluding the robot's view. A trajectory prediction transformer inputs the past observations of the subject (blue dots) and estimates the future position (red dots), which is set as navigation goal for the planner (green line). Once the subject is visible again, the fall is detected. In (b), the robot observes the subject walking in the corridor (blue line). When visual contact is lost, the trajectory prediction (red line) suggests the robot reach the green point to relocate the subject inside the room.

## I. INTRODUCTION

Population aging is one of the most significant challenges for modern societies. By 2050, the number of people aged 65 and older is projected to double [1], increasing the demand for solutions to monitor and support them in maintaining their independence [2], [3]. To address the physical and cognitive decline that comes with aging, service robots [4] have been developed to help people stay healthy and safe at home by assisting with daily tasks and monitoring for potential dangerous events. In such contexts, fall detection is a paramount task, as unintentional falls are the most frequent cause of serious injury among the elderly [5]. Beyond identifying risky situations, trajectory prediction is essential for ensuring continuous people following, as it allows the robot to anticipate a subject's movements and overcome temporary occlusions. However, current approaches still have significant limitations. Fall detection often relies on wearable devices that users do not always accept [6], while vision-based systems struggle to detect subjects or events that aren't fully visible [7]. Similar issues arise in trajectory prediction, particularly when a subject moves outside the robot's field of view. Furthermore, these two tasks are rarely considered together, resulting in a lack of integrated solutions for continuous monitoring.

This paper proposes an innovative framework based on an autonomous mobile platform capable of detecting and following individuals in real-time, predicting their movements, and ensuring a reliable fall detection system by observing the posture evolution over time (Figure 1). The main innovative points are the following:

- A robotic platform is employed for the simultaneous execution of fall detection and trajectory prediction, using a dual-branch transformer architecture with a cross-attention mechanism. An ablation study validates the use of the cross-attention module in connecting the two branches, and demonstrates how the pose encodings from the fall detection branch contribute to improving trajectory prediction.
- High accuracy is achieved for both fall detection and trajectory prediction transformers, surpassing state-of-the-art models as well as classical approaches such as the Kalman filter, particularly in terms of computational efficiency and speed, two fundamental requirements for real-time robotic applications.
- The framework can be applied to any mobile robot equipped with a camera and map-based navigation, responds dynamically to environmental changes, and

[1]Department of Electrical and Information Engineering, Polytechnic University of Bari, IT `a.longo70@phd.poliba.it`
[2]Institute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing (STIIMA), National Research Council of Italy, IT
[3]FieldAI Inc., Irvine CA, USA

offers robust and efficient performance in complex, real-world scenarios. Specifically, a robotic dog is selected, since its use increases social acceptance and enables non-invasive monitoring.

## II. RELATED WORKS

The fall detection task has been addressed in the literature through various technological solutions, which can be classified into sensor-based [3] and vision-based techniques [2]. Sensor-based techniques require users to wear devices, which can limit effectiveness due to non-compliance [8], [6]. In contrast, vision-based approaches are based on fixed cameras and are limited by three main factors: $i$) the need for careful placement within the environment, $ii$) the inability to detect falls when a person is occluded by objects, and $iii$) the restricted, static Field of View (FoV), which cannot dynamically adapt to the movements of a person.

The introduction of mobile robots with specific capabilities represents an interesting novelty in helping older adults live independently [4], since they provide more adaptive and personalized monitoring. Specifically, to perform the fall detection task, mobile platforms must be able to distinguish falls from other ADL to avoid false alarms and follow individuals as they move, maintaining a safe distance and ensuring continuous monitoring even when obstacles temporarily block the camera's FoV. Early methods used geometric measurements to detect falls [9], [10], but these cannot reliably differentiate other activities, such as bending. Recent deep learning approaches exploit mobile platforms for fall detection, using pose estimation with YOLOv8-Pose [11], CNNs with optical flow [12], and YOLOv3 on RGB images [13]. A strong limitation of these works is that the robot does not actually follow the person, but rather moves within previously mapped environments without the ability to adapt dynamically; in addition, the issues of occlusion or the person leaving the robot's FoV remain a significant constraint.

In this context, integrating people-following and trajectory prediction directly into mobile platforms can enhance fall detection task. Transformer-based architectures have emerged as the dominant solution in analyzing data sequences and predicting human movements. For example, [14] introduces POTR, a transformer for single-human motion prediction over short horizons. Building on this, [15] proposes STPOTR (Spatio-Temporal Pose Transformer), which combines spatial and temporal information to improve accuracy and speed in the prediction ahead. Multi-person scenarios are addressed by transformers analyzing human interactions [16], [17] or using spatio-temporal cross-transformers [18] to improve motion prediction. Additionally, transformers can be used to analyze and predict more complex motion patterns [19]. The practical application of these solutions in real-world scenarios, however, remains an active area of research.

User acceptance is crucial for mobile fall detection. Various studies have shown that the effectiveness of such devices depends not only on technical capabilities but also on users' trust [20]. Robotic dogs have emerged as a promising solution, combining mobility and user-friendly design. For instance, [21] presents a robotic dog to support visually blind people, while [22] focuses on using a robotic dog to classify different types of falls. However, a common limitation remains the inability of mobile robots, including robotic dogs, to simultaneously perform fall detection and people-following, especially when the person moves out of the camera's field of view. This highlights the need for novel solutions to address these constraints.

## III. TRANSFORMER-BASED FALL DETECTION AND TRAJECTORY PREDICTION

This work presents a framework that uses a service robot to monitor elderly people. The platform uses onboard sensors of the robot perception module for obstacle-aware navigation while performing people tracking and fall detection. The robot navigation module is in charge of planning the movement of the robotic platform. The core of the proposal is two-fold: $i$) capability to maintain contact with the target and recover from the temporary line of sight loss, and $ii$) perform fall detection in the presence of complex ADL activities. To accomplish both tasks, a dual-branch transformer-based architecture (Figure 2) simultaneously detects falls from a sequence of 2D body landmarks and predicts human trajectory from 3D body motion. The following section describes in detail both branches.

### A. People fall detection

The analysis of literature on deep learning-based fall detection reveals two main challenges: a lack of robustness to scenarios unseen in training data and limitations imposed by a fixed viewpoint. To address this, this paper proposes an approach that uses specific body features to be independent of the camera's viewpoint, along with training the transformer model on multiple datasets to enhance its robustness. The open-source Google MediaPipe [23] has been adopted to extract key body landmarks from the data. To speed up the algorithm, only 11 joints (*front_face, left_shoulder, right_shoulder, left_wrist, right_wrist, left_hip, right_hip, left_knee, right_knee, left_ankle and right_ankle*) of the 33 returned by MediaPipe are retained. Each frame is therefore represented by 11 $< x, y >$ joint pairs, for a total of 22 features. The feature vector is extended with an additional feature, called the Body Aspect Ratio (BAR), that defines the overall orientation of the body. It is computed as the maximum body occupancy in the vertical direction $vd$ over the body width in the horizontal direction $hd$. The fall detection task is modelled as a sequence classification problem, where a sequence of 2D body poses is processed by a transformer model and classified as either a fall or an ADL. The input is first mapped to a higher-dimensional space and passed through two encoder blocks. A pooling operation then aggregates the context of the entire sequence, a series of linear layers and a sigmoid function produce the final classification value.

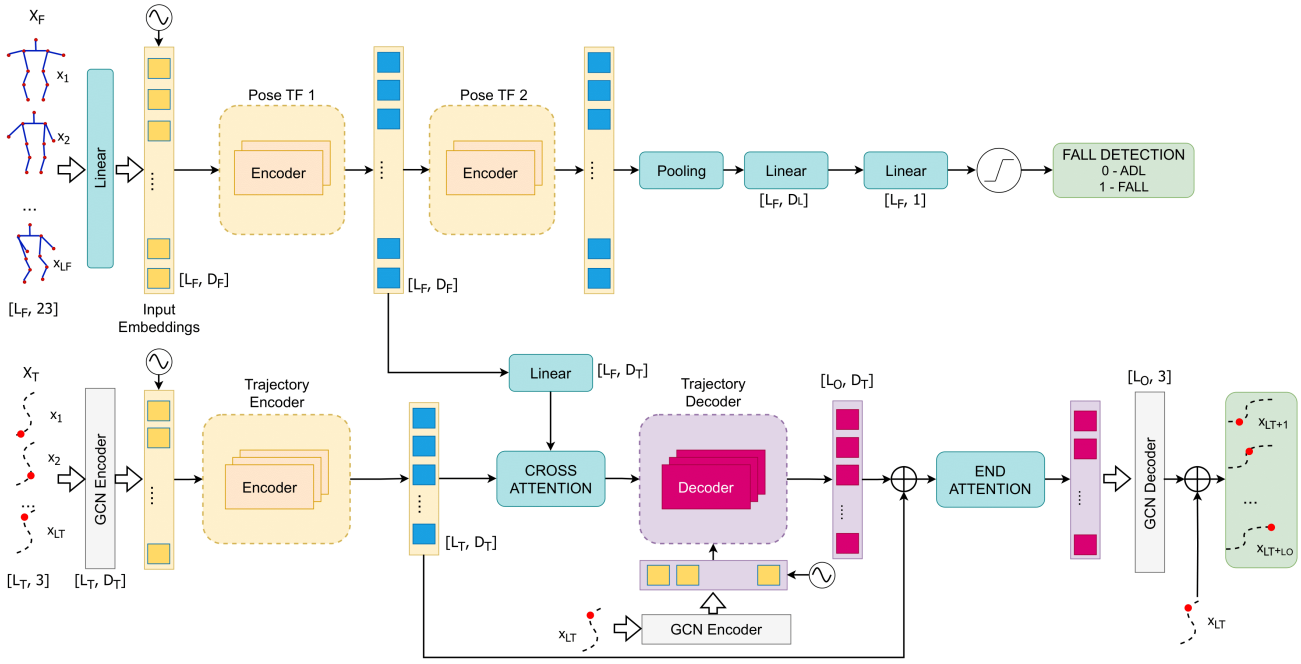



Fig. 2. The proposed dual-branch transformer-based architecture for the combined task of fall detection and trajectory prediction. The upper branch is dedicated to the fall detection activity. A sequence of 2D body landmarks extracted with Google's Mediapipe is the input for the first encoder-only transformer block. The output is processed by a similar block and finally used to classify the sequence as either ADL or Fall. The second branch inputs a sequence of 3D hip positions to predict the future trajectory. The encoder's embeddings are combined with the embeddings of the fall transformer via a cross-attention module. The decoder then receives the output of the cross-attention and the last observation in the input sequence as a query. In this way, the future positions are computed as offsets with respect to the last observed point. Following the decoder, an end attention module, receiving a concatenation of the decoder's output and the encoder's embeddings, is added to capture dependencies in the whole sequence.

### *B. People trajectory prediction*

People following is a fundamental task for mobile robots. Classic vision-based algorithms rely on motion controller stabilization to achieve accurate and responsive behavior in people-following scenarios while maintaining a safe distance from the target. These methods fail in complex indoor environments when the target moves out of sight. To address the above-mentioned limitation, a non-autoregressive transformer for human trajectory prediction is introduced in the navigation stack of the robotic dog (details are reported in the bottom part of Figure 2). The purpose of the model is to estimate the future position of the person under monitoring and use the prediction as a target goal to be reached in case line of sight is lost. This technique thus provides the agent with an initial guess that maximizes the probability of gaining back contact with the target. The robot, therefore, leverages basic vision-based people following and analyzes the human motion to predict its future trajectory and, in case the target goes out of the field of view, moves towards the end-point of the last predicted trajectory. The trajectory prediction problem is handled as a sequence-to-sequence task, where the model analyzes a sequence of past positions to predict future ones. The input embeddings are first computed by a Graph Convolutions Network (GCN) applied to the input hip trajectory and a transformer encoder to extract key features.

A paramount element of the architecture is the cross-attention module, which connects the two branches. The encoder of the fall detection transformer produces latent pose embeddings that capture the overall body dynamics from 2D landmarks. Since these embeddings have a different dimensionality than those derived from the hip trajectory, they are first projected through a fully connected layer to align the representation spaces. The two embeddings, pose and hip, are then fused through the cross-attention mechanism, allowing the trajectory decoder to jointly exploit local motion cues from the hip trajectory and global motion context from the full body pose. This connection ensures that the trajectory prediction branch benefits from complementary features that would not be available from hip motion alone, leading to more accurate and robust trajectory forecasts. Finally, the transformer's decoder, combined with an additional GCN network, generates the future trajectory as a series of offsets from the last observed position.

## IV. EXPERIMENTAL RESULTS

The presented strategy has been implemented on board a mobile legged robot. Below are reported the setup details, along with a description of the obtained results.

### *A. Implementation details*

The core of the proposal is the novel dual-branch transformer for joint human fall detection and trajectory prediction. The tasks are carried out simultaneously, with the addition of a cross-attention module to leverage the features

extracted from 2D poses in the trajectory prediction. Given the different nature of the two tasks, no existing dataset in the literature is well-suited to train the whole architecture at once. Fall detection datasets usually lack 3D motion data, while datasets focusing on 3D trajectories have no falling sequences. Hence, sequential training with a layer-freezing technique has been adopted. The whole process, explained in the following sections, was performed on an Intel Core i7-9700K @ 3.60 GHz processor, with 16 GB RAM, and 24 GB NVIDIA GeForce RTX 4090 GPU.

| | Dataset | Model | Accuracy (%) ↑ | Precision (%) ↑ | Recall (%) ↑ |
|---|---|---|---|---|---|
| [30] | FALL-UP | CNN | 95.64 | 96.91 | 97.95 |
| [28] | FALL-UP | RF | 99.34 | 98.23 | 98.82 |
| [31] | FALL-UP | CNN+LSTM | 98.59 | 91.08 | 94.37 |
| [29] | FALL-UP | SVM | 99.50 | 99.50 | 99.50 |
| [29] | FALL-UP | LSTM | 98.50 | 97.83 | 100 |
| [32] | FALL-UP | CNN+Transformer | 99.55 | - | 98.12 |
| [33] | FALL-UP | GAT+LSTM | 99.54 | 99.70 | 99.03 |
| Ours | FALL-UP | Transformer | 99.71 | 99.31 | 99.65 |
| Ours | Mixed | Transformer | 99.03 | 99.5 | 97.07 |

TABLE I

COMPARISON BETWEEN OUR PROPOSAL AND STATE-OF-THE-ART VISION-BASED FALL DETECTION SYSTEMS USING THE FALL-UP DATASET AND THE PROPOSED MIXED DATASET.

### *B. Fall detection transformer*

The fall detection transformer branch was first trained as an independent architecture. Public datasets often present critical issues regarding their limited size and poor representativeness of real scenarios since they are acquired under controlled conditions. Furthermore, video sequences are generally labelled in a simplified way, neglecting temporal dynamics, thus resulting in a reduction in the reliability of the models trained on such data. To overcome these limitations, the dataset for the experiments was created by integrating several datasets available in the literature, including Le2i [24], UR Fall Detection (URFD) [25], FALL-UP [26] and High-Quality Fall Simulation [27]. The expressiveness of the dataset was improved by splitting each sequence into several sub-sequences of 30 frames each, from which body landmarks were extracted and annotated as a Fall or an ADL. The model was trained for 500 epochs with a learning rate of $10^{-4}$, which is progressively reduced every 50 epochs thanks to an adjustment mechanism. In a preliminary phase, the model was trained and tested exclusively on the FALL-UP dataset, the largest among those considered, to enable a direct comparison with state-of-the-art methods that use the same dataset but different architectures for fall detection. The reference works were selected taking into account the diversity of the approaches, ranging from the more classical machine learning techniques (e.g. Random Forest [28], SVM [29]) to those based on deep learning (e.g. Convolutional Neural Network (CNN) [30], LSTM [29], hybrid models combining CNN and LSTM [31], CNN and Transformer [32], and GAT and LSTM [33]). In Table I, the results demonstrate that the Transformer-based model proposed in this work achieves the highest overall performance on the FALL-UP dataset, outperforming existing methods in terms of accuracy (99.71%), offering a well-balanced trade-off among all evaluation metrics. Furthermore, when tested on the mixed dataset composed of the union of the four datasets, the model maintained high generalization capabilities. Although a slight performance drop occurred compared to single-dataset training, the metrics remained high. This decrease is attributed to the greater heterogeneity of the data, including differences in camera angles, video resolutions, and sensor types, which increase the complexity of the generalization task while making the evaluation more representative of real-world scenarios. Despite this, the results outperform many models trained on individual datasets, demonstrating the robustness of the proposed approach.

In Figure 3, examples of the Transformer model's predictions for fall detection are shown. The model correctly identifies different fall dynamics, such as falling on the knees, and also distinguishes ADL activities, including complex movements like bending, standing up, or leaning without falling. These results demonstrate the model's effectiveness in correctly analyzing activities and discerning borderline situations with complex movements. Regarding the errors (last line in Figure 3), in most cases they occur because the model analyzes limited temporal sequences, which sometimes fail to capture the full dynamics of the action. However, in the subsequent frames, where the action continues, the fall is correctly detected.

### *C. Trajectory prediction transformer*

Upon the complete evaluation of the fall detection branch, the trajectory prediction architecture was added along with the cross-attention module that connects the branches. To train the trajectory prediction module, all the weights of the fall transformer were frozen to avoid any changes during this second training stage. The standard Human3.6M dataset [34] was used for training and testing the model. The dataset contains 3.6 million different 3D articulated poses captured from a set of seven subjects performing 15 activities. While predicting the human trajectory, the fall detection transformer computes the pose embeddings to be

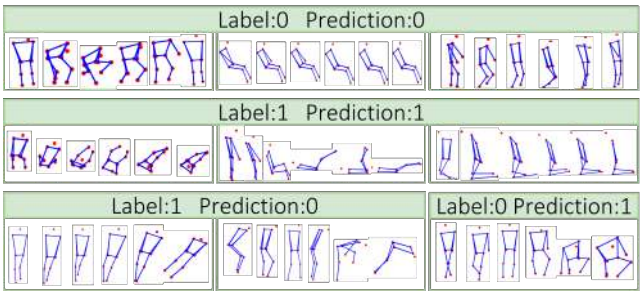



Fig. 3. Example of the transformer model predictions for fall detection. Three different types of ADL ('0') and falls ('1') activities, with their respective ground truth and predicted labels, are reported. Only 6 frames per sequence out of 30 frames are selected for ease of visualization.

fed to the cross-attention module. It inputs the 11 joints that are extracted from 3D poses and puts them in the same 2D image-normalized format as MediaPipe.

The model is trained for 300 epochs, with a learning rate of $10^{-4}$ and a warm-up in the first $10K$ steps. At the end of the second training stage, a unified transformer for the joint task of fall detection and trajectory prediction is obtained. Among the strengths of the proposed architecture is the ability to handle sequences of different lengths. In the proposed settings a sequence length of 30 frames has been used for fall detection, while a sequence length of 5 frames is the trajectory input. Those values are the results of several tests that showed how the fall detection task benefits from longer input sequences to capture complex dynamics (e.g. people lying down and then getting up), while longer input trajectories increase the complexity of the prediction task.

Table II compares the proposed method to the baselines in terms of Average Displacement Error (ADE) and Final Displacement Error (FDE). In the evaluated setting, at 10Hz, the trajectory transformer inputs the past 0.5s observations and predicts the future 2s hip positions. The results show that the model outperforms most existing approaches, achieving an ADE of 0.14 m and a FDE of 0.29 m. Compared to [15], the approach is also competitive with the best-performing published models, obtaining similar accuracy but with a more efficient architecture as highlighted in Table III. The proposed model stands out for its reduced consumption of both the CPU (4.21% on average) and the GPU (6.30% on average), compared to [15], which shows a higher usage, especially of the GPU (83.34% on average). Furthermore, the inference time was reduced to 0.004 seconds, almost half of the time required by the baseline model. Another strength is the architectural simplicity with 267 layers and approximately 3.3 million parameters, than the 382 layers and over 17 million parameters of [15], making it more suitable for real-time scenarios.

Figure 4 shows some qualitative results on trajectory predictions, highlighting a very low displacement error, especially in the first chunks of the predicted sequence. Moreover, despite a higher error in the final predicted point, the model successfully predicts the overall motion direction of the subject, which is the most important thing for the work.

In addition, the proposed transformer-based approach was compared with classical trajectory prediction methods, namely the Kalman Filter. The Kalman Filter with velocity estimation via Least Squares introduced in [39] was adapted to 3D trajectories. The filter parameters were tuned to achieve the best performance for the proposed use case. The filter dynamically adapts to the system noise using a displacement-based model with window size $e = 3$ and coefficients $a = 0.05$, $b = 10^{-4}$. Velocity was estimated using linear regression over a window of $l = 5$ steps. From each filtered state, the next 20 positions were predicted under the assumption of constant velocity. Results given in Table IV showcase that the transformer achieves significantly lower ADE and FDE, highlighting its ability to model complex, non-linear motion patterns and temporal dependencies that are challenging for linear filtering approaches.

| | Prediction length (s) | ADE(m) ↓ | FDE(m) ↓ |
|---|---|---|---|
| [35] | 2.0 | 0.42 | 0.51 |
| [36] | 2.0 | 0.41 | 0.50 |
| [37] | 2.0 | 0.36 | 0.47 |
| [38] | 2.0 | 0.35 | 0.44 |
| [15] | 2.0 | 0.13 | 0.27 |
| Ours | 2.0 | 0.14 | 0.29 |

TABLE II

COMPARISON BETWEEN OUR PROPOSAL AND STATE-OF-THE-ART SYSTEMS FOR TRAJECTORY PREDICTION ON THE HUMAN3.6M DATASET.

| | CPU (%) ↓ | | GPU (%) ↓ | | Time (s) ↓ | Layers ↓ | Parameters ↓ |
|---|---|---|---|---|---|---|---|
| | AVG | PEAK | AVG | PEAK | | | |
| [15] | 6.37 | 7.90 | 83.34 | 92.00 | 0.007 | 382 | 17,902,999 |
| Ours | 4.21 | 5.40 | 6.30 | 11.4 | 0.004 | 267 | 3,380,424 |

TABLE III

COMPARATIVE EVALUATION OF PERFORMANCE BETWEEN [15] AND OUR MODEL, CONSIDERING THE CPU AND GPU USAGE, AS AVERAGE AND PEAK VALUES, THE INFERENCE TIME, THE TOTAL NUMBER OF LAYERS, AND THE OVERALL NUMBER OF MODEL PARAMETERS.

### D. Ablation Study

The ablation study of the proposed model (Figure 2) was conducted to evaluate the impact and effectiveness of its different modules. The architecture is composed of two transformer-based branches: the first dedicated to fall detection and the second to trajectory prediction. Based on [15] and motivated by the assumption that the pose encodings produced by the fall detection module can enhance trajectory prediction, several strategies can connect the two transformer branches. Among these, a cross-attention module and a feature concatenation technique were employed. In the first experiment, the results obtained without the cross-attention module were compared with those obtained when the module was included. This module takes as inputs the encodings of the trajectory branch and the output of the second encoder of the fall detection transformer. The results (Table V) indicate that using this module improves

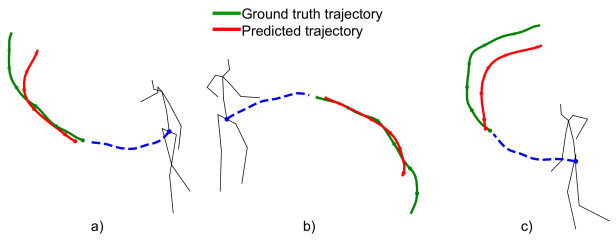



Fig. 4. Trajectory prediction results on the Human3.6M dataset. Starting from the hip trajectories (blue line), the model predicts the future trajectory (red line), which is compared to the ground truth trajectory (green line).

| Method | ADE(m) ↓ | FDE(m) ↓ |
|---|---|---|
| Kalman Filter | 0.49 | 0.62 |
| Ours | 0.14 | 0.29 |

TABLE IV

COMPARISON BETWEEN OUR PROPOSAL AND THE KALMAN FILTER APPROACH FOR TRAJECTORY PREDICTION.

both ADE and FDE. To further improve performance, particularly in terms of FDE, an alternative approach was tested exploiting a feature concatenation technique. In this approach, the same inputs used for the cross-attention module are concatenated, and the resulting vector is passed through linear layers to project it to a dimension compatible with the decoder of the trajectory prediction branch. This computationally lighter method was introduced to determine whether the complexity of the cross-attention module was a limiting factor or not. However, since no significant improvements were observed, an alternative strategy was adopted to enhance trajectory prediction, using the output of the first encoder instead of the second. The encodings were too abstract and heavily influenced by fall detection information, whereas those from the first encoder were less abstract and better represented the evolution of the pose. These encodings were used both as input to the cross-attention module and for feature concatenation. The results reported in Table V show that the best performance was achieved using the output of the first encoder in combination with the cross-attention module. These findings suggest that leveraging intermediate-level pose representations from the fall detection branch can effectively enhance trajectory prediction. Although the cross-attention module introduces additional complexity in the model, the performance gain justifies its inclusion.

| Transformer Branch | Encoder Output | Connection Method | ADE(m) | FDE(m) |
|---|---|---|---|---|
| Trajectory | None | None | 0.18 | 0.36 |
| Trajectory + Fall Detection | 2nd encoder | Cross-Attention | 0.16 | 0.30 |
| Trajectory + Fall Detection | 2nd encoder | Feature Concatenation | 0.19 | 0.36 |
| Trajectory + Fall Detection | 1st encoder | Cross-Attention | 0.14 | 0.29 |
| Trajectory + Fall Detection | 1st encoder | Feature Concatenation | 0.18 | 0.34 |

TABLE V

ABLATION STUDY RESULTS COMPARING DIFFERENT CONNECTION STRATEGIES BETWEEN THE TWO TRANSFORMER-BASED BRANCHES AND THE ENCODER OUTPUTS FROM THE FALL DETECTION TRANSFORMER FOR TRAJECTORY PREDICTION.

### *E. Real hardware setup*

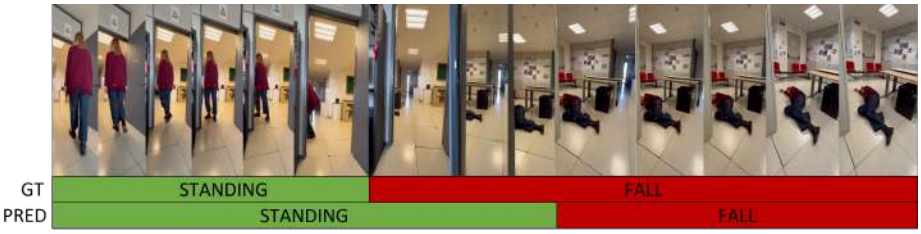
The system has been tested on the legged platform GO2 from Unitree Robotics, which is equipped with an NVIDIA Jetson Orin on-board computer, running Ubuntu 20.04 operating system. The robot perception module relies on a HESAI XT16 LiDAR and an Intel RealSense D435i depth camera. The robot navigation module is given by the onboard SLAM system, LIO-SAM [40], which employs the LiDAR stream to build and maintain an updated map of the environment. In parallel, the people detection module inputs the camera stream from the Intel RealSense D435i and generates the 2D joints' coordinates. A projection function from the RealSense SDK is also used to detect the hip 3D position with respect to the camera reference system. All this information is then used to simultaneously get the probability of a fall event and the future trajectory of the moving subject. In case the subject has moved out of the robot's field of view, the last predicted position is transformed into map absolute coordinates and sent as navigation goal to the robot navigation module. This module exploits the built-in planner to generate an obstacle-aware trajectory first, and low-level commands to move the robot towards the target waypoint.

### *F. Real world experiments*

Several experiments were performed in a real-world environment with static and dynamic obstacles. Evaluating the reliability of the fall detection transformer requires careful consideration of borderline cases, which represent potential points of failure. These cases are characterized by complex ADL sequences in which the subject's actions may result in false alarms because of misidentification as a fall, which may compromise the system's reliability. Figure 5(a) shows that the subject bends down to tie the shoe and then stands up to walk. In this case, a static analysis would generate an alarm because the images of the subject on the ground are similar to examples of falls. In contrast, the strength of a system that can analyze temporal sequences lies in its ability to extract information from the entire sequence. As the results demonstrate, the transformer-based model correctly predicts the case as a non-fall through a complete analysis of the pose of the subject who, after bending, gets up to walk. In addition, as illustrated in Figure 1, the most challenging situations occur when the robot loses sight of a person who falls behind an obstacle or enters another room. In these cases, the robustness of the trajectory module becomes paramount. With reference to Figure 1(a), the trajectory prediction transformer uses past observations (blue dots) to estimate the future position (red dots) of the subject behind the table. Consequently, even when the robot loses the target, the on board planner, based on the prediction, is able to compute a path (green line) to reach and find the subject. Once the subject returns to the robot's field of view, the fall detection branch correctly detects that a fall has occurred, as shown in Figure 5(b). The second example, shown in Figures 1(b) and 5(c), illustrates a situation where the robot, while maintaining a safe distance from the subject, loses sight of the person when the subject turns behind a door. In this case, while the vision-based follow me stops as there is a lack of visible references, the trajectory transformer suggests that the subject may be behind the door. Following this indication, the robot navigates to the predicted waypoint, finding the subject again and allowing the system to correctly detect the fall.


(a)


(b)




(c)

Fig. 5. Real-world experiments acquired by the onboard camera of the robotic dog. In (a), a challenging ADL activity is shown, in which the subject walks, bends down, and then stands back up. In (b) and (c), two examples are presented in which the robot loses sight of the person. In the first case (b), the subject walks and then falls behind an obstacle, while in the second one (c), the subject turns behind a door before falling. In both, the robot uses the trajectory prediction module to successfully locate and identify the person's position. The bottom bar in all the subfigures shows the comparison between the ground truth label (GT) and the model prediction (PRED).

## V. CONCLUSIONS

This work proposed an integrated model on a robotic dog platform, based on a dual-branch transformer architecture designed for elderly assistance. The system combines fall detection with trajectory prediction, providing advanced support to improve users' safety and autonomy. The model achieved satisfactory results in terms of activity classification and trajectory prediction when tested on both public and real-world data. Tests demonstrate that the mobile platform can track people around obstacles, detect falls even in occluded areas, and predict trajectories, thus ensuring a continuous body view. Another key strength of the proposed solution is its compatibility with any mobile robot that has a camera and a map-based navigation system. Interesting perspectives involve extending the transformer model to consider a multi-class action recognition approach and managing the presence of multiple subjects in the scene. The identification of multiple actions could enable the classification of different behaviors such as falling, sitting, and standing, enhancing the monitoring of elderly people's habits. The observation of several subjects, their recognition in cases of overlap within the camera's field of view, and their interaction with the environment would not only enable simultaneous tracking of people but also improve the robot's navigation performance.

## REFERENCES

[1] D. o. E. United Nations and P. D. Social Affairs, "World population ageing 2019," 2020, sT/ESA/SER.A/444. [Online]. Available: https://digitallibrary.un.org/record/3907988

[2] L. M. Dang, K. Min, H. Wang, M. J. Piran, C. H. Lee, and H. Moon, "Sensor-based and vision-based human activity recognition: A comprehensive survey," *Pattern Recognition*, vol. 108, p. 107561, 2020.

[3] F. Luna-Perejon, J. Civit-Masot, I. Amaya-Rodriguez, L. Duran-Lopez, J. P. Dominguez-Morales, A. Civit-Balcells, and A. Linares-Barranco, "An automated fall detection system using recurrent neural networks," in *Artificial Intelligence in Medicine: 17th Conference on Artificial Intelligence in Medicine, AIME 2019, Poznan, Poland, June 26–29, 2019, Proceedings 17*. Springer, 2019, pp. 36–41.

[4] P. Asgharian, A. M. Panchea, and F. Ferland, "A review on the use of mobile service robots in elderly care," *Robotics*, vol. 11, no. 6, p. 127, 2022.

[5] W. H. Organization, "Ageing and life course unit," *WHO global report on falls prevention in older age*, 2008.

[6] C. A. U. Hassan, F. K. Karim, A. Abbas, J. Iqbal, H. Elmannai, S. Hussain, S. S. Ullah, and M. S. Khan, "A cost-effective fall-detection framework for the elderly using sensor-based technologies," *Sustainability*, vol. 15, no. 5, p. 3982, 2023.

[7] R. Igual, C. Medrano, and I. Plaza, "Challenges, issues and trends in fall detection systems," *Biomedical engineering online*, vol. 12, no. 1, p. 66, 2013.

[8] R. Steele, A. Lo, C. Secombe, and Y. K. Wong, "Elderly persons' perception and acceptance of using wireless sensor networks to assist healthcare," *International journal of medical informatics*, vol. 78, no. 12, pp. 788–801, 2009.

[9] A. Tomoya, S. Nakayama, A. Hoshina, and M. Sugaya, "A mobile robot for following, watching and detecting falls for elderly care," *Procedia computer science*, vol. 112, pp. 1994–2003, 2017.

[10] A. Elwaly, A. Abdellatif, and Y. El-Shaer, "New eldercare robot with path-planning and fall-detection capabilities," *Applied Sciences*, vol. 14, no. 6, p. 2374, 2024.

[11] S. U. Ahamad, M. Ataei, V. Devabhaktuni, and V. Dhiman, "Omobot: a low-cost mobile robot for autonomous search and fall detection," in *2024 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2024, pp. 453–460.

[12] C. Menacho and J. Ordoñez, "Fall detection based on cnn models implemented on a mobile robot," in *2020 17th international conference on ubiquitous robots (UR)*. IEEE, 2020, pp. 284–289.

[13] S. Lafuente-Arroyo, P. Martín-Martín, C. Iglesias-Iglesias, S. Maldonado-Bascón, and F. J. Acevedo-Rodríguez, "Rgb camera-based fallen person detection system embedded on a mobile platform," *Expert Systems with Applications*, vol. 197, p. 116715, 2022.

[14] A. Martínez-González, M. Villamizar, and J.-M. Odobez, "Pose transformers (potr): Human motion prediction with non-autoregressive transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 2276–2284.

[15] M. Mahdavian, P. Nikdel, M. TaherAhmadi, and M. Chen, "Stpotr: Simultaneous human trajectory and pose prediction using a non-autoregressive transformer for robot following ahead," *arXiv preprint arXiv:2209.07600*, 2022.

[16] P. Xiao, Y. Xie, X. Xu, W. Chen, and H. Zhang, "Multi-person pose forecasting with individual interaction perceptron and prior learning," in *European Conference on Computer Vision*. Springer, 2024, pp. 402–419.

[17] J. Wang, H. Xu, M. Narasimhan, and X. Wang, "Multi-person 3d motion prediction with multi-range transformers," *Advances in Neural Information Processing Systems*, vol. 34, pp. 6036–6049, 2021.

[18] H. Yu, X. Fan, Y. Hou, W. Pei, H. Ge, X. Yang, D. Zhou, Q. Zhang, and M. Zhang, "Toward realistic 3d human motion prediction with a spatio-temporal cross-transformer approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 10, pp. 5707–5720, 2023.

[19] Y. Mao, J. Deng, W. Zhou, Y. Fang, W. Ouyang, and H. Li, "Masked motion predictors are strong 3d action representation learners," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 10 181–10 191.

[20] P. Asgharian, A. M. Panchea, and F. Ferland, "A review on the use of mobile service robots in elderly care," *Robotics*, vol. 11, no. 6, 2022.

[21] A. Bazhenov, V. Berman, S. Satsevich, O. Shalopanova, M. A. Cabrera, A. Lykov, and D. Tsetserukou, "Dogsurf: Quadruped robot capable of gru-based surface recognition for blind person navigation," in *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, 2024, pp. 238–242.

[22] S. Liu, T. Zhang, H. Ji, and L. Wang, "A novel yolov8-pose-based algorithm for abnormal behavior analysis of quadruped robots," in *2024 IEEE 13th Data Driven Control and Learning Systems Conference (DDCLS)*. IEEE, 2024, pp. 1538–1543.

[23] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, *et al.*, "Mediapipe: A framework for building perception pipelines," *arXiv preprint arXiv:1906.08172*, 2019.

[24] I. Charfi, J. Miteran, J. Dubois, M. Atri, and R. Tourki, "Optimised spatio-temporal descriptors for real-time fall detection: comparison of svm and adaboost based classification," *Journal of Electronic Imaging (JEI)*, vol. 22, no. 4, p. 17, 2013.

[25] B. Kwolek and M. Kepski, "Human fall detection on embedded platform using depth maps and wireless accelerometer," *Computer methods and programs in biomedicine*, vol. 117, no. 3, pp. 489–501, 2014.

[26] L. Martínez-Villaseñor, H. Ponce, J. Brieva, E. Moya-Albor, J. Núñez-Martínez, and C. Peñafort-Asturiano, "Up-fall detection dataset: A multimodal approach," *Sensors*, vol. 19, no. 9, p. 1988, 2019.

[27] G. Baldewijns, G. Debard, G. Mertes, B. Vanrumste, and T. Croonenborghs, "Bridging the gap between real-life data and simulated data by providing a highly realistic fall dataset for evaluating camera-based fall detection algorithms," *Healthcare technology letters*, vol. 3, no. 1, pp. 6–11, 2016.

[28] H. Ramirez, S. A. Velastin, I. Meza, E. Fabregas, D. Makris, and G. Farias, "Fall detection and activity recognition using human skeleton features," *Ieee Access*, vol. 9, pp. 33 532–33 542, 2021.

[29] T. Y. Koffi, Y. Mourchid, M. Hindawi, and Y. Dupuis, "An improved 3d skeletons up-fall dataset: enhancing data quality for efficient impact fall detection," in *Seventeenth International Conference on Machine Vision (ICMV 2024)*, vol. 13517. SPIE, 2025, pp. 215–222.

[30] R. Espinosa, H. Ponce, S. Gutiérrez, L. Martínez-Villaseñor, J. Brieva, and E. Moya-Albor, "A vision-based approach for fall detection using multiple cameras and convolutional neural networks: A case study using the up-fall detection dataset," *Computers in biology and medicine*, vol. 115, p. 103520, 2019.

[31] A. R. Inturi, V. Manikandan, and V. Garrapally, "A novel vision-based fall detection scheme using keypoints of human skeleton with long short-term memory network," *Arabian Journal for Science and Engineering*, vol. 48, no. 2, pp. 1143–1155, 2023.

[32] B. Li, J. Li, and P. Wang, "Fall detection algorithm based on global and local feature extraction," *Pattern Recognition Letters*, vol. 185, pp. 31–37, 2024.

[33] B. Kim, J. Im, and B. Noh, "Federated learning-based road surveillance system in distributed cctv environment: Pedestrian fall recognition using spatio-temporal attention networks," *Applied Intelligence*, vol. 55, no. 6, pp. 1–16, 2025.

[34] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, 12 2013.

[35] Y. Yuan and K. Kitani, "Dlow: Diversifying latent flows for diverse human motion prediction," in *European Conference on Computer Vision*. Springer, 2020, pp. 346–364.

[36] D. Wei, H. Sun, B. Li, J. Lu, W. Li, X. Sun, and S. Hu, "Human joint kinematics diffusion-refinement for stochastic motion prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 5, 2023, pp. 6110–6118.

[37] M. Jiang and L. Hu, "Efficient human motion prediction in 3d using parallel enhanced attention with sparse spatiotemporal modeling," *Electronics*, vol. 14, no. 9, p. 1773, 2025.

[38] H. J. Kim and E. Ohn-Bar, "Motion diversification networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 1650–1660.

[39] M. Kamezaki, M. Hirayama, R. Kono, Y. Tsuburaya, and S. Sugano, "Human velocity estimation using kalman filter and least squares with adjustable window sizes for mobile robots," *IEEE Access*, 2024.

[40] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2020, pp. 5135–5142.