

Article

Guessing Human Intentions to Avoid Dangerous Situations in Caregiving Robots

Noé Zapata , Gerardo Pérez , Lucas Bonilla , Pedro Núñez * , Pilar Bachiller  and Pablo Bustos 

RoboLab, Robotics and Artificial Vision, University of Extremadura, 10003 Cáceres, Spain;
nzapata@unex.es (N.Z.); gperezgonz@unex.es (G.P.); lubonillar@unex.es (L.B.); pilarb@unex.es (P.B.);
pbustos@unex.es (P.B.)

* Correspondence: pnuntru@unex.es

Abstract: The integration of robots into social environments necessitates their ability to interpret human intentions and anticipate potential outcomes accurately. This capability is particularly crucial for social robots designed for human care, as they may encounter situations that pose significant risks to individuals, such as undetected obstacles in their path. These hazards must be identified and mitigated promptly to ensure human safety. This paper delves into the artificial theory of mind (ATM) approach to inferring and interpreting human intentions within human–robot interaction. We propose a novel algorithm that detects potentially hazardous situations for humans and selects appropriate robotic actions to eliminate these dangers in real time. Our methodology employs a simulation-based approach to ATM, incorporating a “like-me” policy to assign intentions and actions to human subjects. This strategy enables the robot to detect risks and act with a high success rate, even under time-constrained circumstances. The algorithm was seamlessly integrated into an existing robotics cognitive architecture, enhancing its social interaction and risk mitigation capabilities. To evaluate the robustness, precision, and real-time responsiveness of our implementation, we conducted a series of three experiments: (i) A fully simulated scenario to assess the algorithm’s performance in a controlled environment; (ii) A human-in-the-loop hybrid configuration to test the system’s adaptability to real-time human input; and (iii) A real-world scenario to validate the algorithm’s effectiveness in practical applications. These experiments provided comprehensive insights into the algorithm’s performance across various conditions, demonstrating its potential for improving the safety and efficacy of social robots in human care settings. Our findings contribute to the growing research on social robotics and artificial intelligence, offering a promising approach to enhancing human–robot interaction in potentially hazardous environments. Future work may explore the scalability of this algorithm to more complex scenarios and its integration with other advanced robotic systems.

Keywords: social robotics; cost-effective; artificial theory of mind; human–robot interaction



Citation: Zapata, N.; Pérez, G.; Bonilla, L.; Núñez, P.; Bachiller, P.; Bustos, P. Guessing Human Intentions to Avoid Dangerous Situations in Caregiving Robots. *Appl. Sci.* **2024**, *14*, 8057. <https://doi.org/10.3390/app14178057>

Academic Editors: Alessandro Umbrico and Daniel Beßler

Received: 24 July 2024

Revised: 23 August 2024

Accepted: 27 August 2024

Published: 9 September 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent decades, robotics have advanced by leaps and bounds, opening up new possibilities in various areas, including human care and assistance. Assistive robots, or “caregiving robots”, have emerged as a promising solution to address the challenges associated with an aging population and caregiver shortages. These robots are designed to help the elderly, disabled, or chronically ill by providing physical, emotional, and social support. To facilitate human–robot collaboration, these robots must be sophisticated in their social environment, demonstrating advanced reactivity, interaction, and comprehension [1]. The development of social awareness in robots is a complex process influenced by many factors, including cultural norms, social cues, diverse human activities, and the individual preferences of those in their vicinity. To achieve social adeptness, robots must go beyond the mere perception of their physical surroundings. They must develop the ability to

comprehend any situation's social and contextual nuances. This involves, among other things, recognizing the presence of people in the robot's surroundings and interpreting their behaviors, emotions, and intentions within the broader social context [2].

Caregiving robots, in particular, face unique challenges that demand an even higher level of social intelligence. These robots must be able to continuously sense and interpret the ongoing activities of individuals within their operational environment. This real-time analysis serves a crucial purpose: anticipating potential risk scenarios before they materialize. Armed with this foresight, caregiving robots must be able to adjust their behavior swiftly and effectively to mitigate emerging risks in a manner that is effective and socially acceptable. The achievement of these goals requires the integration of several advanced capabilities. Firstly, caregiving robots must have predictive algorithms to forecast forthcoming events. Secondly, they need sufficient computational power to analyze these predictions rapidly and foresee their potential consequences across various scenarios. Finally, and perhaps most challenging, these robots must be programmed with an understanding of human social norms and expectations. This understanding should guide the robot's decision-making process, allowing it to select actions that effectively address the immediate situation and align with societal expectations of appropriate behavior.

In this paper, we expand upon previous research in the field of artificial theory of mind (ATM) [3] and its relationship to simulation-based internal models [4,5]. Theory of mind (TM) is a fundamental cognitive capability that enables individuals to predict their actions and those of others [6]. This concept provides a plausible framework for understanding how humans can anticipate the behavior of others in specific situations [4,7]. Within the broader field of TM research, several theoretical approaches have been proposed to elucidate the underlying cognitive processes. In this study, we adopt the simulation theory (ST) variant, which posits that "... people use imagination, mental pretense, or perspective-taking ('putting oneself in the other person's shoes') to determine the mental states of others" [8]. This approach emphasizes the role of mental simulation in understanding and predicting the thoughts, feelings, and actions of others.

Our primary contribution in this paper is twofold: (i) We aim to further support the hypothesis that simulation-based internal models can serve as a viable foundation for implementing ATM in artificial systems. By demonstrating the efficacy of this approach, we seek to bridge the gap between human cognitive processes and artificial intelligence systems; (ii) Building upon this theoretical framework, we present a working model and a practical example of human caregiving in social robots. This model showcases the application of ATM principles in a concrete, real-world scenario, illustrating how simulation-based approaches can enhance the social intelligence of robotic systems.

2. Related Work

The ATM in robotics has evolved significantly since its inception, playing a novel role in developing socially intelligent robots. This section provides an overview of the key developments in ATM, highlighting the progression from early conceptual frameworks to current state-of-the-art approaches.

Scassellati's seminal work laid the foundations of ATM in robotics [3], which proposed and implemented a detailed framework in the humanoid robot Cog. This pioneering effort catalyzed research in the domain, establishing ATM as a crucial component in developing socially intelligent robots. Building on this foundation, Kennedy et al. [7] presented an early implementation of simulation-based ATM using an embodied version of ACT-R/E. Their "like-me" approach demonstrated the potential of enhancing a robot's perceptual capabilities, particularly in inferring human gaze direction. However, this early work did not extend to modeling these inferences as intentions or translating them into actionable plans, a limitation our current research aims to address.

Gray et al. [9] explored a robot's ability to manipulate human beliefs by concatenating action-simulation and mental-state-simulation primitives. While their work presented a scheme for recursively modeling beliefs, it focused on illustrating how embodiment

connects agents' mental states rather than applying ATM to social robots in real-time scenarios. A recent work [4,5] introduced an architecture featuring a simulation-based internal model called the consequence engine. Although their approach shares similarities with ours, their robots operated in a 2D grid environment, with simulated actions being projections of their state into the future using existing movement controllers. In contrast, our model assigns intentions to people as potential relations with visually detectable objects, which are then transformed into actions, allowing for a more abstract representation conducive to adaptation and learning.

Lemaignan et al. [10] proposed an architecture with an ATM module where simulation is carried out by a high-level planner using shared plans. Their approach extends ATM into a theory theory (TT) variation, differing from the internal model-based simulator approach [11]. Recent studies have explored the relationship between ATM in robots and its effect on people's acceptance or trust [12], as well as its connection with ethics and security [13]. The findings of empirical studies indicate that human beings are more inclined to accept, and place greater trust in, robots that have been programmed with the capacity for TM abilities [14]. Recent advancements in deep learning and reinforcement learning have also contributed to the field of ATM in robotics. For example, Rabinowitz et al. [15] introduced a neural network model called theory of mind network (ToMnet) that can learn to model the mental states of other agents from observations of their behavior.

From a broader perspective, detecting human intentions has gained significant interest in social robotics, particularly in socially aware robot navigation [16]. For instance, Ferrer et al. [17] developed a system where the robot accompanies a person while predicting their intentions to reach a destination. Kostavelis et al. [18] proposed a system that detects human intentions regarding goal positions, considering places with semantic meaning as potential targets. Mavrogiannis et al. [19] presented a system that detects signals of intentions or preferences regarding avoidance strategies, enabling the agent to simplify people's decision-making. Skrzypczyk et al. [20] introduced a control scheme for an intelligent wheelchair navigation assistant that detects cooperative and non-cooperative behaviors by comparing predicted pedestrian positions to real ones. While we share with these works the goal of characterizing intentions, our experiment also explored how intentions are converted into actions and their potential consequences. Shvo et al. [21] presented an algorithm for proactive assistance in robots. It integrates epistemic planning-based techniques and adds TM capabilities, with the aim of inferring human plans from the environment and executing a task based on the inferred plan. Cunningham et al. [22] proposed an approach more aligned with ours, where the robot simulates itself and other agents forward under assigned policies to obtain predicted states and observation sequences. However, their work differs in how context is represented and how new goals (e.g., dangerous situations) are detected and integrated into the search for successful actions.

Related to detecting collision risks, Koide et al. [23] predicted both trajectories and dangerous situations using a convolutional neural network (CNN). In this approach, a local environmental map is used as an input, and the CNN determines whether a collision risk exists and predicts the human trajectory. The training of the CNN is conducted using datasets of tracked individuals, where the authors incorporated the introduction of virtual obstacles along the trajectory of the individuals. This idea is based on the assumption that if a person is not aware of an object in his or her immediate vicinity, that person will behave as if the object is not there. This concept has significant parallels with ours for planning a human's path to a desired target. Our research extends to the notion that the robot is positioned to observe an obstacle in close proximity, and thus how this can facilitate the avoidance of a hazardous situation. Moreover, the utilization of a CNN as an opaque entity that determines whether a human is at risk of collision entails, in the case of an erroneous network output (i.e., false positive), an inability to identify the underlying causes of the unsatisfactory result and, consequently, the capacity to act by enhancing the detector's functionality.

3. Methodology

This section describes the model that assigns intentions to humans using the same cognitive mechanisms our robot employs to attach itself to local goals. This approach, known as a “like-me” simulation [7], leverages the robot’s capabilities to understand and predict human behavior. Our robot, Shadow [24], has advanced visual object detection and navigation capabilities. It can identify objects in its environment, select them as targets, and navigate toward them along obstacle-free paths. We extend this functionality to interpreting human intentions by establishing conceptual links between nearby humans and objects in their vicinity.

The robot interprets these human–object links as potential actions that the person might perform, limited to the actions the robot can execute. This constraint ensures that the robot’s predictions remain within the realm of its experiential framework, enhancing the accuracy of its simulations. To comprehend the consequences of human actions, the robot employs an internal model to simulate these actions from a third-person perspective. This approach aligns with the theory proposed in [25], which posits that a robot understands a phenomenon if it possesses a model for it and the logical implications within that model correspond to real-world causality.

To explore this concept further, we equipped the robot with a simplified model of itself and its interactions with the environment. This model is executed within an internal physics-based simulator (PyBullet), allowing the robot to explore the potential consequences of its actions within the simulator’s possibilities (e.g., collisions between elements) and develop executable plans. Extending this reasoning, we apply our understanding theory to the intentions our model assigns to humans. The robot simulates possible human actions, evaluating their environmental consequences and potential security risks. If a simulated intention poses a risk to an individual, the robot continues its simulation process, exploring whether any actions could mitigate or eliminate the identified risk. If a viable solution is found, the robot executes the risk-mitigating action.

This process is illustrated in Figure 1, which depicts a scenario where a person is assigned two potential targets: a door and a couch. Each target generates several possible trajectories. When one of these trajectories is identified as dangerous (e.g., an obstacle in the path), the robot simulates an action that could avert the potential risk. The robot conducts a secondary simulation to validate the effectiveness of its planned intervention. This simulation places the robot at the imagined final position of its proposed action and re-simulates the person’s trajectory. This process allows the robot to confirm that the person would recognize the additional physical presence of the robot and consequently plan a new, obstacle-free path.

3.1. The CORTEX Architecture

For our experiments, we employed a streamlined version of the CORTEX robotics cognitive architecture [26]. CORTEX is a multi-agent architecture designed to facilitate the creation of information flows among various memories and modules. These flows are driven by agents, which share their information thanks to the working memory \mathcal{W} . \mathcal{W} is a distributed graph structure implemented using conflict-free replicated data types (CRDT) [27]. It can be formally defined as

$$\mathcal{W} = (V, E, A_V, A_E) \quad (1)$$

where V is the set of vertices (nodes) representing elements of a predefined ontology, $E \subseteq V \times V$ is the set of edges, $A_V : V \rightarrow 2^{\text{Attr}}$ is a function mapping vertices to sets of attributes, and $A_E : E \rightarrow 2^{\text{Attr}}$ is a function mapping edges to sets of attributes. Edges can encode geometric transformations or logical predicates. Both nodes and edges can store a list of attributes of predefined types.

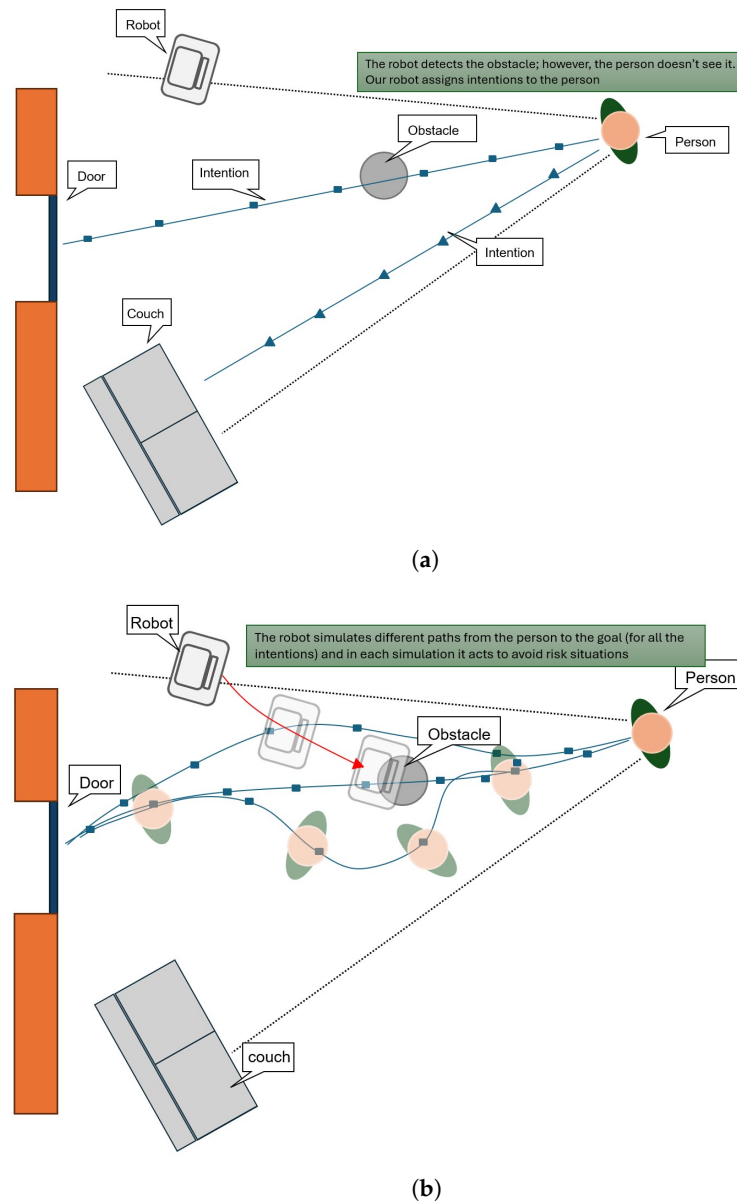


Figure 1. The robot identifies two potential targets for the human: a door and a couch. For each target, it generates a trajectory, excluding any obstacles that fall outside the human's field of view from the planning process. Upon analysis of the generated trajectories, a potential collision with an obstacle is identified in the trajectory corresponding to the door. To avoid this collision and maintain the human's safety, the robot considers a range of potential actions that could be taken to eliminate the collision in the human's trajectory towards the door. (a) Estimation of trajectories from the person to the objects of interest. (b) Possible collision avoidance action, displacement of the robot to the obstacle.

Figure 2 illustrates the CORTEX cognitive architecture, depicting its hierarchical structure across cognitive and sub-cognitive levels. The system is anchored by our central \mathcal{W} at the cognitive level, represented as an octagonal shape. \mathcal{W} contains representations of various elements in the robot's environment, including 'robot', 'person', 'couch', 'intention', and 'door'. Surrounding \mathcal{W} there are several CORTEX agents (e.g., 'Mem' (Memory): This connects to a more significant 'Memories' box above, which includes semantic, episodic, procedural, and spatial memory types; 'Sim' (Simulation): This links to an 'Inner simulator' component, visualized with abstract shapes suggesting mental modeling or prediction capabilities; 'Body': Interfacing with the sub-cognitive level; and 'Atom' (Atomic): Purpose not explicitly defined). In Figure 2, the sub-cognitive level contains a network of inter-

connected software components that handle lower-level processes (e.g., Sensory inputs: ‘LiDAR(s) 3D’ and ‘Camera 360’; Processing modules: ‘Tracker’, ‘Instance’, ‘Semantic’, ‘Grid’, ‘MPC’ (model predictive control), and ‘Bumper’; Action output: ‘Omni base’ for movement; and ‘Controller’ coordinating these sub-components). The sub-cognitive level takes in visual objects and regions (VE) as input and outputs move_to(e) commands to a visual target.

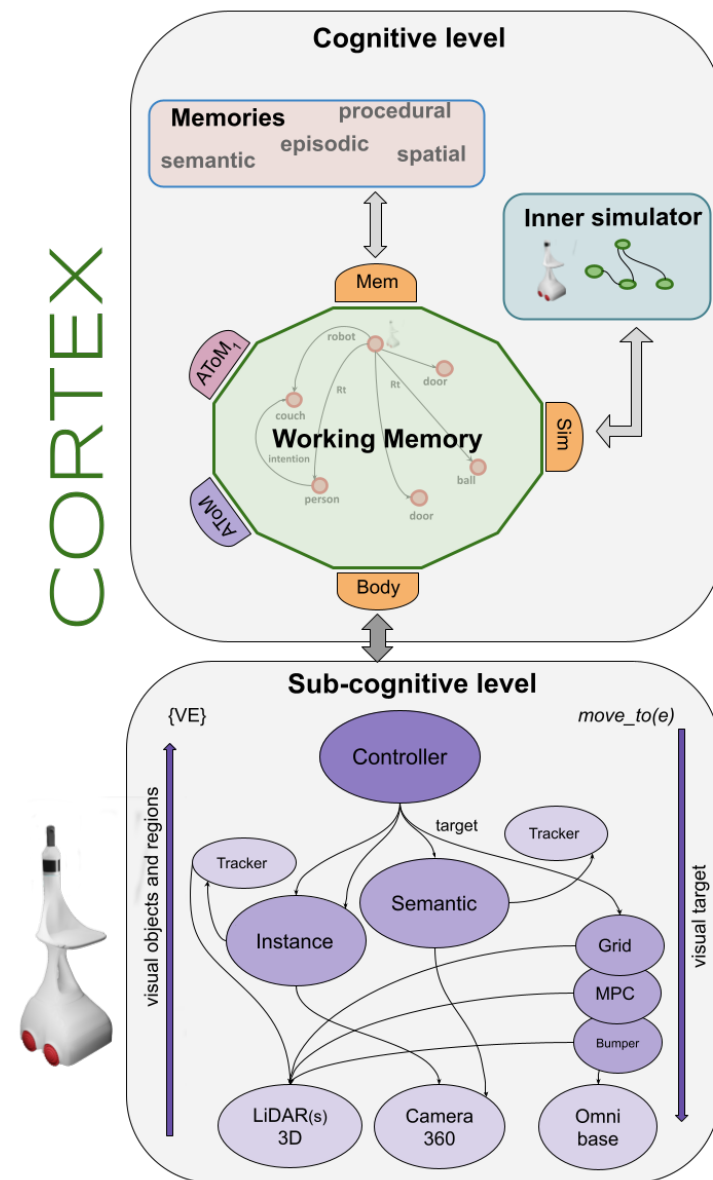


Figure 2. Schematic representation of the CORTEX architecture. The architecture is divided into two levels: cognitive and sub-cognitive. The cognitive level focuses on hosting \mathcal{W} and the internal simulator (PyBullet), while the sub-cognitive level hosts the software that manages the low-level processes.

The cognitive CORTEX architecture has been successfully used in different robotic platforms [28]. In our simulation-based approach to ATM, the essential elements of CORTEX are identified as follows:

- \mathcal{W} , as defined above.
- The sub-cognitive level, S , which can be represented as a function:

$$S : I \rightarrow \mathcal{W} \times T \quad (2)$$

where I is the input from sensors, and T is the set of possible visual targets for navigation.

- The software agent A_S connecting the sub-cognitive module with \mathcal{W} :

$$A_S : S \rightarrow \mathcal{W} \quad (3)$$

- The internal simulator Σ , based on PyBullet (<https://pybullet.org/> accessed on 22 August 2024), and its interfacing agent A_Σ :

$$\Sigma : \mathcal{W} \times A \rightarrow \mathcal{W}' \quad (4)$$

where A is the set of possible actions, and \mathcal{W}' is the simulated future state of \mathcal{W} .

- Two new agents, A_{ATM1} and A_{ATM2} , implementing the artificial theory of mind (ATM):

$$A_{ATM1} : \mathcal{W} \rightarrow \mathcal{W} \times I \quad (5)$$

$$A_{ATM2} : \mathcal{W} \times I \rightarrow \mathcal{W} \times A \quad (6)$$

where I is the set of inferred intentions, and A is the set of possible actions.

Agents can modify \mathcal{W} to create and maintain an updated context, making it accessible to all other agents. The information flow can be described as a series of transformations:

$$\mathcal{W}_t \xrightarrow{A_S} \mathcal{W}_{t+1} \xrightarrow{A_{ATM1}} (\mathcal{W}_{t+2}, I) \xrightarrow{A_{ATM2}} (\mathcal{W}_{t+3}, A) \quad (7)$$

where \mathcal{W}_t represents the state of \mathcal{W} at time t .

3.2. Simulation-Based Approach to Artificial Theory of Mind

We present our novel approach to the ATM in caregiving robots. We leverage a “like-me” simulation paradigm to assign intentions to humans and predict their actions within the robot’s operational environment. This model enables the robot to utilize its internal representations to comprehend and anticipate human behavior, enhancing its social intelligence and risk mitigation capabilities.

3.2.1. Problem Formulation

\mathcal{W} contains all current robot beliefs about its environment. The context ctx is defined as the content of \mathcal{W} , including all the robot’s current beliefs about its environment. The internal simulator typically runs synchronously with the context, effectively maintaining a copy of the state. When the “like-me” process starts, the simulator is temporarily detached, and people’s intentions and the robot’s actions can be safely simulated. Upon completion, the simulator goes back into synchronization mode.

We define an intention as the engagement that a person establishes with an object in the environment, possibly through its affordances. Intentions are enacted by linking them to one or more of the robot’s potential actions and then simulated with its internal model.

Let $H = \{h_1, h_2, \dots, h_n\}$ be the set of humans in the environment and $O = \{o_1, o_2, \dots, o_m\}$ be the set of detectable objects.

We introduce an intention function $I : H \times O \times A \times G \rightarrow \{0, 1\}$, where A is the set of possible actions and G is the set of possible gaze directions. $I(h_i, o_j, a_k, g_l) = 1$ if the intention is considered risky (i.e., leads to a collision), and 0 otherwise.

We have implemented two algorithms that read and modify \mathcal{W} . The first agent is activated when people are in \mathcal{W} . It will guess and enact their intentions, marking some of them as dangerous if a collision is detected. The marked intentions activate the second agent, seeking a feasible robot intervention that cancels it. Both agents communicate through annotations in \mathcal{W} . Two predefined lists are provided to simplify the notation: Gaze and Actions, which contain the minimum and maximum values for the vertical gaze of a generic human and the list of the robot’s actions, respectively.

3.2.2. Intention Assignment and Simulation

According to the ctx information, for each person $h_i \in H$ and object $o_j \in O$ within the person's field of view, we generate intentions for each possible action $a_k \in A$ and sampled gaze direction $g_l \in G$:

$$int_{ijkl} = intention(h_i, o_j, a_k, g_l) \quad (8)$$

Each intention is then simulated:

$$c = sim(ctx, int_{ijkl}) \quad (9)$$

where c is a flag indicating whether a collision occurred during the simulation. Algorithm 1 formalizes the process of intention guessing and enacting.

Algorithm 1 starts by accessing the current context ctx provided by \mathcal{W} and the lists of people H and objects O in the robot's environment. For each person, h_i , the list of objects entering their field of view ($Tgts$) is obtained using a simple geometric inclusion test over a predefined frustum (line 5). Each object is assumed to be a potential interaction target for the person. Thus, for each person and target, intentions enacted by potential actions must be simulated to predict risky situations. An intention is considered dangerous if its enactment entails a collision. Since the risk assessment depends on the person's ability to perceive the situation, the person's gaze is also considered during the simulation. We assume the robot cannot estimate this from its position, so the predefined range in $Gaze$ is uniformly sampled (line 8). According to the person (p_i), the object (t_j), the action (act_k), and the gaze ($gaze_l$), an intention int_{ijkl} is finally generated (line 9).

Algorithm 1 Intention guessing and enacting

Require: $W, Gaze, Actions$

Ensure: W

```

1:  $ctx \leftarrow getContext(W)$ 
2:  $People \leftarrow getPeople(ctx)$ 
3:  $Objects \leftarrow getObjects(ctx)$ 
4: for all  $p_i \in People$  do
5:    $Tgts \leftarrow fov(p_i, Objects)$ 
6:   for all  $t_j \in Tgts$  do
7:     for all  $act_k \in Actions$  do
8:       for all  $gaze_l \in sample(Gaze)$  do
9:          $int_{ijkl} \leftarrow intention(p_i, t_j, act_k, gaze_l)$ 
10:         $c \leftarrow sim(ctx, int_{ijkl})$ 
11:         $W \leftarrow addToW(W, (int_{ijkl}, c))$ 
12:       end for
13:     end for
14:   end for
15: end for

```

The simulator is then called to execute that intention given the current context (ctx) (line 10). This call implies freezing the current context so changes can be made without interfering with the perception of reality. The simulation proceeds by executing action act_k under the constrained access to objects in the scene given by $gaze_l$. In this experiment, the robot's only action is $goto(x)$, so the simulator proceeds in two steps: (a) a path-planning action to compute a safe route from the person to the target object, in which the occupancy grid is modified to include only the objects in the person's field of view; and (b) the displacement of the person along that path in the unmodified scene. The first step corresponds to the question "how does the person move through her environment given the assigned gaze?" and the second to "how does the robot imagine from its point of view what the person will do?" The path is executed using a copy of the robot's path-following controller. The simulation provides the answers to the two questions, returning a flag c signaling the

occurrence of a collision. Finally, in line 11, \mathcal{W} is updated to include the new intention with the attribute c , possibly marking a risky situation.

3.2.3. Intervention Planning

Algorithm 2 describes the action selection process for intervention. For each risky intention int_{ijkl} (where $c = 1$), the robot considers its own possible actions $a_m \in A$ and objects $o_n \in O$:

$$robot_{mn} = intention(robot, o_n, a_m) \quad (10)$$

The robot then simulates the combination of its action and the human's intention:

$$c = sim(ctx, [robot_{mn}, int_{ijkl}]) \quad (11)$$

If $c = 0$ (no collision), the robot's action $robot_{mn}$ is considered a valid intervention.

Algorithm 2 Action selection

Require: W , $Actions$

Ensure: W

```

1:  $ctx \leftarrow getContext(W)$ 
2:  $Intentions \leftarrow getIntentions(ctx)$ 
3:  $Objects \leftarrow getObjects(ctx)$ 
4: for all  $int_{ijkl} \in Intentions$  where  $int_{ijkl} \in Risk$  do
5:   for all  $act_m \in Actions$  do
6:     for all  $o_n \in Objects$  do
7:        $robot_{mn} \leftarrow intention(robot, o_n, act_m)$ 
8:        $c \leftarrow sim(ctx, [robot_{mn}, int_{ijkl}])$ 
9:       if  $\neg c$  then
10:         $W \leftarrow addToW(W, robot_{mn})$ 
11:       end if
12:     end for
13:   end for
14: end for
```

Algorithm 2 runs on a different agent and initiates when an intention is marked as risky in \mathcal{W} . The algorithm iterates over the robot's action set, $Actions$, to check if executing any of them removes the risky intention. There is no a priori knowledge of which action to try first or if any of them will succeed. The first loop goes through the intentions marked as dangerous (line 4). Then, following a similar procedure as in Algorithm 1, the robot assigns to itself intentions to all visible objects. The robot's intention $robot_{nm}$ along with the person's intention int_{ijkl} are sent for simulation (line 8). This call entails the re-enactment of int_{ijkl} , but this time also considering the robot's action. If that action makes the collision disappear, it is selected for execution in the real world by adding the corresponding robot intention to \mathcal{W} .

4. Experimental Results

This section describes the experiments carried out to evaluate our proposal. Initially, we describe the scenario used, both in the simulation and in the real experiments. Then, we detail the three specific tests conducted for the evaluation.

4.1. The Scenario

As a test bed for our research, we created a scenario using the Webots simulator. Figure 3 shows the simulated scenario, which represented a room with the following elements: an autonomous robot; a person positioned near the wall opposite the door; a door and a couch, which represent potential targets for the person; and a soccer ball in the center of the room, which adds complexity as a moving obstacle that can cause a dangerous

situation if the person does not catch sight of it. In this simulation, the robot had to use its internal model to predict the person's possible trajectories, acting proactively to avoid collisions and ensure safety.

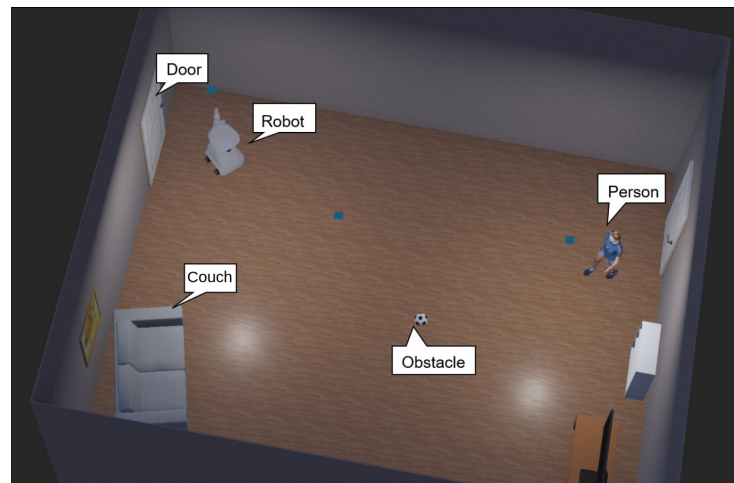


Figure 3. Simulated scenario used to evaluate the robot's ability to anticipate and mitigate risks. The room contains several key elements: an autonomous robot positioned near the door, a person located near the wall opposite the door, a door and a couch representing potential targets for the person, and a soccer ball as a dangerous object during human movement. The robot must use its internal model to predict possible trajectories of the person, acting proactively to ensure safety.

The robot was a custom-built unit named Shadow [24], with a rectangular base with four Mecanum wheels. The robot's main sensors were a 360° camera placed on a 3D LiDAR. Both devices were situated in the uppermost part and were co-calibrated. The 3D pointcloud was projected onto the camera image to provide a sparse depth plane that was used to estimate the distance to regions in the image. All objects in the scene were recognizable by the you only see once (YOLO) (we used YoloV8 from Ultralytics. <https://www.ultralytics.com/> accessed on 22 August 2024) deep neural network [29]. There were no other distractors in the room. If the detected element was a person, its skeleton was passed to a second DNN, JointBDOE (<https://github.com/hnuzhy/jointbdoe> accessed on 22 August 2024), to estimate their orientation [30]. All detected objects were assigned a depth coordinate obtained from the LiDAR. In addition, they were tracked using the ByteTrack algorithm [31] to assign them a persistent identification tag. We defined a visual element e as belonging to the set of YOLO recognizable objects VE . The stream of detected visual elements was defined as $e_t \in \{VE\}$, where $t \geq 0$. The robot in this scenario could only execute one action, $move_to(e_i)$, where e_i is a visually detectable object in the scene. It did not have a speaker. The action was executed by a local controller consisting of several components: a path planner over a local grid, a model predictive control over the generated trajectory, and a virtual bumper [24]. No global map was needed to execute the visually guided motions. The robot had a known maximum speed and acceleration. A new target $e \in \{VE\}$ was sent to the local controller whenever the robot was required to approach a different object in the scene.

4.2. Validation of the Approach

The two proposed algorithms were implemented as new agents in the architecture and tested with three experiments. The first one consisted of a series of simulations in Webots, where some free parameters were sampled. The second introduced uninformed human subjects in the loop to evaluate their reactions to the robot intervention, and the third experiment was performed with a real robot and person.

These experiments were designed to test the ATM model and its implementation as a means of detecting potentially dangerous situations caused by people's intentions. We

assumed that people always chose the risky path. Thus, we did not try to predict and detect other possible choices.

4.2.1. Situation Analysis and Decision-Making Test

The first experiment consisted of 180 initializations of the nominal scenario, letting the robot analyze the situation and decide which action to take. The sampled variables were the positions of the robot, the ball, and the person. The range of positions used in all cases was a 1.5 m radius. The human always followed a direct trajectory towards the couch at a constant speed of 0.5 m/s. Figure 4 shows the different elements involved in the experiment. The graph in the upper-left shows the current state of \mathcal{W} with the robot node labeled Shadow at the center. The perceived objects and the person were connected to the robot with RT edges that stored an $SE(2)$ geometric transformation. The rest of the edges were 2-place logic predicates representing perceived relationships between nodes. In this example, a `has_intention` edge connects the person to their intended destination, and a `collision` node signals that they may implement their intention via an unsafe path. At the top right, the contents of \mathcal{W} are graphically displayed. The person is represented by a yellow circle connected to a green cone, which limits their field of view. Two paths connect the person to the couch. One of the paths crosses the obstacle, represented as a red box, indicating a possible collision. The couch is drawn as a green box. The robot is depicted twice, in red occupying its starting position and in light green occupying the imagined position that will make the person change their path. The lower left image is a zenithal view of the scene, and the right image shows the PyBullet scene used to simulate the robot and the person traversing the paths.

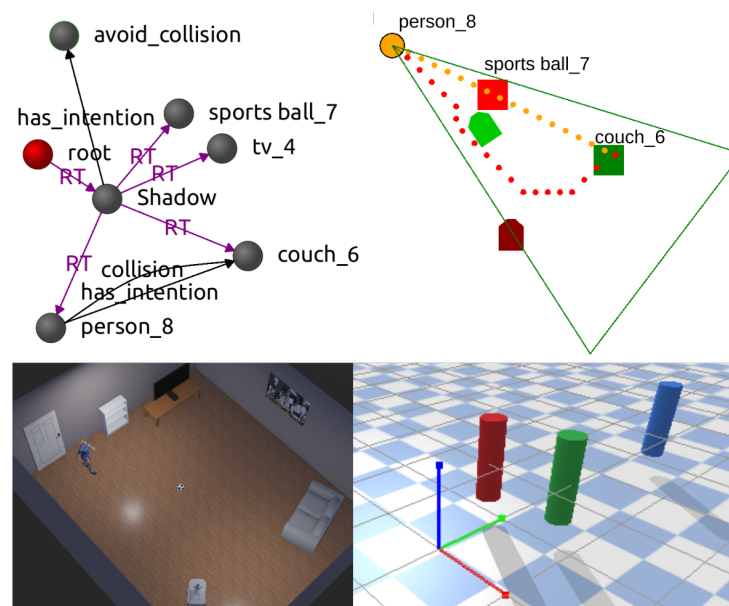


Figure 4. Combined view showing the contents of \mathcal{W} (**up-left**); a 2D graphical representation of \mathcal{W} with two paths going from the person to the couch (**up-right**), the yellow path represents the route that would take the person directly to the couch without seeing the obstacle, while the red path shows the alternative route when the robot is moved; a zenithal view of the scene as rendered by Webots (**down-left**); and a 3D view of the internal simulator, PyBullet, with simple geometric forms representing the elements in the scene (**down-right**).

Table 1 shows the results of 180 runs with the Webots simulator. Of the total number of experiments, 13 were discarded due to the absence of detected intentions. Considering the remaining experiments (167), the proposed intention-guessing algorithm provided an accuracy rate of 79.64%. The primary source of loss of accuracy was the detection of false positives. This means that the robot wrongly detected some risky situations that would,

in reality, not pose a danger to the person. The primary causes of these false positives were the inaccurate sizes of the shapes representing the person and the obstacles in the internal simulator, as well as the errors in the estimations of real distances due to noise in the LiDAR measurements. Nevertheless, the algorithm did not produce false negatives, i.e., all the unsafe intentions were correctly identified by the robot and no risky situation was left unattended. Both facts can also be derived from the resulting precision and recall values. Given the consequences of failing to detect jeopardized intentions, recall can be considered more vital than precision. Taking this into consideration, we measured F1 and F2 scores [32]. While the obtained F1 score was moderately high, reaching a value of approximately 0.76, the F2 score indicated a high level of effectiveness.

Regarding the time metrics, the mean reaction time, which is the period from detecting a possible collision to the appearance of an action that can avoid it, was less than 0.75 s. The experiment suggested that an under one second response time is sufficient for real-time operation in this scenario. Additionally, the robot could generate an action that prevented the person from colliding in most situations detected as risky. Although the reaction time will grow with the number of objects, people, and actions in the scene, with their corresponding free parameters, given some of the heuristics and pruning strategies discussed in the next section, we believe it can still be kept low enough to work in more complex scenarios.

Table 1. Summary of experimental results.

Metric	Value
Total Experiments	180
Experiments Discarded	13
Valid Experiments	167
Accuracy Rate	79.64%
False Positives	20.35%
False Negatives	0%
Precision	0.61
Recall	1.0
F-1 Score	0.76
F-2 Score	0.89
Reaction time (mean)	747 ms
Reaction time (deviation)	310 ms
Experiments with Detected Collision and Action Generated	85.18%

4.2.2. Human-in-the-Loop Test

A second experiment was conducted to test the reaction to the robot's intervention of human subjects unfamiliar with the problem. This experiment had a human-in-the-loop configuration, where six subjects were instructed to control the person's trajectory in the simulator using a joystick and a view from a virtual camera placed on an avatar. The subjective camera was oriented in a way that prevented the subjects from seeing the ball on the floor. The subject's goal was to reach the couch on the other side of the room, and they were only allowed to play with the joystick for a few seconds. After completing the experiment, all subjects avoided the hidden ball when the robot entered their field of view and was detected. Qualitatively, all subjects generated different trajectories with unequal free margins to the unseen obstacle but completed the task without incidence. These variations were not considered relevant to the experiment. Figure 5 shows a sequence of six frames from one of these trials. The series runs from top left to bottom right, and each frame is split into two: the Webots zenithal view of the scene at the top and the subjective view shown to the human subject at the bottom. In the sequence, as the robot enters the field of view, the person corrected the trajectory and drove the avatar to the target position.

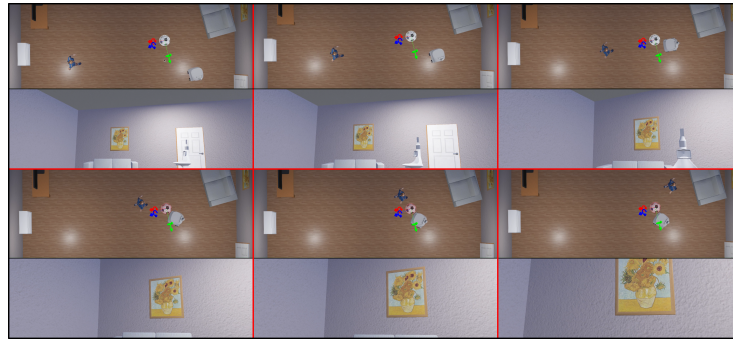


Figure 5. Human-in-the-loop experiment (top left to bottom right). The upper half of each frame is the zenithal view rendered by the Webots simulator, where the red, blue, and green axes mark the reference system of the obstacle between the person and the couch. The lower half of each frame is representative of the first-person perspective, as observed by the human subject who was undertaking the experimental procedure. As can be seen, the obstacle was outside the field of view. When the robot approached the obstacle, frames 2–3, the subject turned left, overcoming it and safely reaching the couch.

4.2.3. Real-World Test

The third experiment was performed with the Shadow robot [24] in a real scenario. A group of five subjects were instructed to cross the room while reading a paper and heading towards a chair on the other side. A backpack was in the way (We replaced the football with a backpack because, in the real scenario, YOLO detected it much more confidently), and they were instructed to ignore it. They were not briefed about the robot participation. In all five trials, the robot detected the subject and moved close to the unseen obstacle to prevent the person from stumbling. The subjects reacted to the robot's intervention with varying degrees of surprise due to the unexpectedness of the event, but all continued on their way to the assigned target. This variety in people's reactions could be used to adapt the internal model according to a comfort metric, resulting in smoother, less startling movements. This idea is commented on in the next section. Figure 6 shows a three-frame sequence from one of the trials. The lower part of each frame shows a graphic representation of \mathcal{W} with the robot in brown heading upwards, the person in yellow heading downwards, the backpack in red, and the chair in green. The line of yellow dots shows the path attributed to the subject during the internal simulation phase.

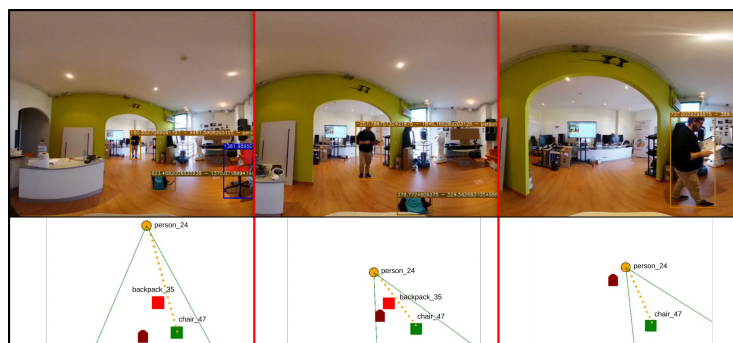


Figure 6. Real-world experiment (left to right). The upper half of the frame shows the view from the robot's camera. The lower half shows a schematic view of \mathcal{W} , with the person represented as a yellow circle, the backpack on the floor as a red square, and the target chair as a green square. The robot is colored dark red. The subject walks distractedly towards the chair (frame 1) and reacts when the robot starts moving (frame 2), changing direction and continuing.

5. Discussion

The ATM is slowly gaining momentum as a tool for adding third-view reasoning capabilities to robotics control architectures, but there are still many open issues. Our

research shows that an internal physics-based simulator embedded in a robot's control architecture could be the key to integrating the ATM into the set of cognitive abilities. Current technology allows direct and precise control of a simulator's scene graph and time step. However, this integration relies on the existence of a \mathcal{W} offering a stable and updated view of the current context based on a hybrid numeric–symbolic representation. This memory relies, in turn, on other modules that provide the necessary connections to the world. Putting all these together implies a considerable development effort and, as we see it, the ability to further pursue this task is linked to three key aspects related to the feasibility of the approach: (a) the stability and reliability of \mathcal{W} representing the current context in which inferences can be made; (b) the combinatorial explosion caused by the nested relationships between participants and the free parameters of their actions; and (c) the robustness required for reliable real-time execution in more open scenarios.

The multi-agent CORTEX architecture addresses the first issue to a reasonable extent, with \mathcal{W} being updated by agents that connect to the subcognitive modules. Although there is ample room for improvement, the results show that the representation is stable and can handle a real-world, real-time situation under fairly realistic conditions. In addition, the underlying data structure, which holds symbolic and numeric values, can be monitored to provide a clear view of the internal reasoning process, facilitating a path to accountability and ethics.

A second problem is the exponential growth caused by the nested searches that unfold all possible relationships among the robot, its actions, the objects in the scene, and the individuals. This search can be curtailed if we consider ways to exit the loops as soon as possible or even to skip them altogether. Studies in humans suggest that no more than two or three alternative strategies are evaluated, at least without interrupting the ongoing behavior and resorting to more extended reflection [33,34]. A first prune could be achieved by forcing time and space constraints. In our experiments, the person-to-collision time limited the admissibility of a robot's action, i.e., displacements that will not arrive in time are discarded. Another limitation comes from the maximum number of people that can be represented in \mathcal{W} simultaneously, which is an architecture parameter. This number could be further reduced by distance to the robot, relative position, or certain attributes of a person that could be relevant in the current context. A further reduction could be achieved if the search ends as soon as an action is found that removes the danger, but more efficiently if the sets are ordered according to some metric or drawn from an a priori distribution learned from experience. For example, intentions could be assigned to objects according to prior knowledge of the person's habits, and actions could be assigned to intentions biased by the objects' affordances.

The third aspect, robustness, is particularly relevant for robot operation in human-populated scenarios. Robustness depends on the stability of the representation, on the ability to learn from experience the optimal values of actions' and objects' free parameters, i.e., robot speed, human habits, specific place to stop, time to start, etc., in each context, and on the ability to recover from failures and unforeseen events during the execution. This suggests that to improve the robustness, the control architecture has to be more adaptable at each level of its organization, integrating learning algorithms that capture the effect of the interventions on humans in specific contexts to fine-tune the free parameters.

An interesting issue arises in complex real-world scenarios when more than one person is in the scene and they have assigned risk intentions. In this case, the robot would face the dilemma of saving one and ignoring the other. This kind of decision would involve a more complex evaluation of the situation, using a system of values and thus entering the emerging field of social robot ethics [35].

In a simulated environment, the controlled space allows efficient development and testing of algorithms. A simple scenario was modeled where the robot interacted with a single person, a fixed obstacle, and a target object. The sensor data were augmented with computed virtual noise to closely mimic real-world conditions. This approach ensured that the simulated sensor data, particularly from LiDAR, closely approximated the noise char-

acteristics and variability found in real-world measurements, thereby producing distance measurements that were comparable across both environments. This method enhanced the realism of the simulation, allowing the robot to process sensor data in a way that closely mirrored real-world scenarios.

Furthermore, the algorithm was designed to identify and classify objects using a YOLO neural network, which inherently makes the system dependent on the network's output. The performance of this neural network could vary significantly between simulated and real environments due to differences in data quality and variability. Moreover, while in a simulation a person follows a predefined or random trajectory within a limited range, allowing the robot to easily anticipate the future position of the person and the obstacle using the internal simulation, in the real world, people can behave unexpectedly or irrationally. This unpredictability adds an additional layer of complexity to conducting real experiments, requiring monitoring mechanisms that continuously assess different variables, from ongoing hazard assessment to the feasibility of the action taken. This enables the robot to halt unnecessary actions that may disrupt the human's actions.

6. Conclusions and Future Work

The work presented in this paper is a new step towards an ATM that can be effectively run inside a robotics cognitive architecture. We rely on a \mathcal{W} that holds a representation stable enough for an inner simulator to take third-view perspectives.

In future work, we expect to increase the number and complexity of the robot's actions, while keeping the number of combinations small, using algorithms that learn an optimal selection order from experience, anytime design techniques, or sorting heuristics from external knowledge sources.

One potential avenue for future research would be to extrapolate the proposed algorithm to other known robotic systems. The employment of \mathcal{W} permits all agents to be concurrently aware of changes to this shared memory space. Given the limited number of agents currently included in the system, it may be feasible to extrapolate the communication between these agents through \mathcal{W} to a publish/subscribe system between agents. However, as the architecture has been designed in a modular way, an increase in the robot's ability to infer human actions and act accordingly may result in a corresponding increase in the complexity of inter-agent communication.

In the context of this investigation, given that the robot was solely equipped with a base actuator to undertake corrective actions in response to a dangerous scenario, the “like-me” simulation was conducted through the simulation of GOTO actions. The incorporation of supplementary interaction mechanisms into the robot's operational capabilities would facilitate a significant expansion in the range of scenarios that could be subjected to analysis. This, in turn, would enable the formulation of a multitude of actions that could effectively mitigate the risk of adverse situations involving diverse interaction mechanisms.

Author Contributions: Conceptualization, N.Z., P.B. (Pilar Bachiller), P.B. (Pablo Bustos) and P.N.; methodology, N.Z. and G.P.; software, G.P., L.B. and N.Z.; validation, G.P., N.Z. and P.B. (Pablo Bustos); formal analysis, N.Z. and P.N.; investigation, N.Z. and P.B. (Pilar Bachiller); resources, N.Z.; data curation, N.Z. and G.P.; writing—original draft preparation, N.Z. and G.P.; writing—review and editing, P.N. and P.B. (Pablo Bustos); visualization, L.B.; supervision, P.N.; project administration, P.N.; funding acquisition, P.B. (Pablo Bustos) and P.N.; All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially funded by TED2021-131739-C22, supported by Spanish MCIN/AEI/10.13039/501100011033 and the European Union's NextGenerationEU/PRTR, by the Spanish Ministry of Science and Innovation PDC2022-133597-C41 and by FEDER Project 0124 EU-ROAGE MAS 4 E (2021–2027 POCTEP Program).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy and ethical restrictions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Mahdi, H.; Akgun, S.A.; Saleh, S.; Dautenhahn, K. A survey on the design and evolution of social robots—Past, present and future. *Robot. Auton. Syst.* **2022**, *156*, 104193. [CrossRef]
2. Tewari, M.; Lindgren, H. Expecting, understanding, relating, and interacting-older, middle-aged and younger adults' perspectives on breakdown situations in human–robot dialogues. *Front. Robot. AI* **2022**, *9*, 956709. [CrossRef]
3. Scassellati, B. Theory of Mind for a Humanoid Robot. *Auton. Robot.* **2002**, *12*, 13–24. [CrossRef]
4. Winfield, A.F.T. Experiments in Artificial Theory of Mind: From Safety to Story-Telling. *Front. Robot. AI* **2018**, *5*, 75. [CrossRef] [PubMed]
5. Blum, C.; Winfield, A.F.T.; Hafner, V.V. Simulation-Based Internal Models for Safer Robots. *Front. Robot. AI* **2018**, *4*, 74. [CrossRef]
6. Carruthers, P.; Smith, P.K. *Theories of Theories of Mind*; Cambridge University Press: Cambridge, UK, 1996.
7. Kennedy, W.; Bugajska, M.; Harrison, A.; Trafton, J. “Like-Me” Simulation as an Effective and Cognitively Plausible Basis for Social Robotics. *Int. J. Soc. Robot.* **2009**, *1*, 181–194. [CrossRef]
8. Shanton, K.; Goldman, A. Simulation theory. *WIREs Cogn. Sci.* **2010**, *1*, 527–538. [CrossRef] [PubMed]
9. Gray, J.; Breazeal, C. Manipulating Mental States Through Physical Action. *Int. J. Soc. Robot.* **2014**, *6*, 315–327. [CrossRef]
10. Lemaignan, S.; Warnier, M.; Sisbot, E.A.; Clodic, A.; Alami, R. Artificial Cognition for Social Human-Robot Interaction: An Implementation. *Artif. Intell.* **2017**, *247*, 45–69. [CrossRef]
11. Simulation Theory Versus Theory Theory: Theories Concerning the Ability to Read Minds. Master's Thesis, Leopold-Franzens-Universität Innsbruck, Innsbruck, Austria, 2002.
12. Rossi, A.; Andriella, A.; Rossi, S.; Torras, C.; Alenyà, G. Evaluating the Effect of Theory of Mind on People's Trust in a Faulty Robot. In Proceedings of the 2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), Napoli, Italy, 29 August–2 September 2022; pp. 477–482, ISSN 1944-9437. [CrossRef]
13. Vanderelst, D.; Winfield, A. An architecture for ethical robots inspired by the simulation theory of cognition. *Cogn. Syst. Res.* **2017**, *48*, 56–66. [CrossRef]
14. Ruocco, M.; Mou, W.; Cangelosi, A.; Jay, C.; Zanatto, D. Theory of Mind Improves Human's Trust in an Iterative Human-Robot Game. In Proceedings of the 9th International Conference on Human-Agent Interaction, New York, NY, USA, 9–11 November 2021; HAI '21; pp. 227–234. [CrossRef]
15. Rabinowitz, N.C.; Perbet, F.; Song, H.F.; Zhang, C.; Eslami, S.A.; Botvinick, M. Machine theory of mind. *arXiv* **2018**, arXiv:1802.07740.
16. Singamaneni, P.T.; Bachiller-Burgos, P.; Manso, L.J.; Garrell, A.; Sanfeliu, A.; Spalanzani, A.; Alami, R. A Survey on Socially Aware Robot Navigation: Taxonomy and Future Challenges. *arXiv* **2023**, arXiv:2311.06922. [CrossRef]
17. Ferrer, G.; Garrell, A.; Herrero, F.; Sanfeliu, A. Robot social-aware navigation framework to accompany people walking side-by-side. *Auton. Robot.* **2017**, *41*, 775–793. [CrossRef]
18. Kostavelis, I.; Kargakos, A.; Giakoumis, D.; Tzovaras, D. Robot's Workspace Enhancement with Dynamic Human Presence for Socially-Aware Navigation. In Proceedings of the ICVS, Shenzhen, China, 10–13 July 2017; Liu, M., Chen, H., Vincze, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2017; Volume 10528, pp. 279–288.
19. Mavrogiannis, C.I.; Thomason, W.B.; Knepper, R.A. Social Momentum: A Framework for Legible Navigation in Dynamic Multi-Agent Environments. In Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction, New York, NY, USA, 5–8 March 2018; HRI '18; pp. 361–369. [CrossRef]
20. Skrzypczyk, K. Game Against Nature Based Control of an Intelligent Wheelchair with Adaptation to Pedestrians' Behaviour. In Proceedings of the 2021 25th International Conference on Methods and Models in Automation and Robotics (MMAR), Międzyzdroje, Poland, 23–26 August 2021; pp. 285–290. [CrossRef]
21. Shvo, M.; Hari, R.; O'Reilly, Z.; Abolore, S.; Wang, S.Y.N.; McIlraith, S.A. Proactive Robotic Assistance via Theory of Mind. In Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 23–27 October 2022; pp. 9148–9155. [CrossRef]
22. Cunningham, A.; Galceran, E.; Mehta, D.; Ferrer, G.; Eustice, R.; Olson, E., MPDM: Multi-policy Decision-Making from Autonomous Driving to Social Robot Navigation. In *Control Strategies for Advanced Driver Assistance Systems and Autonomous Driving Functions*; Springer: Cham, Switzerland, 2019; pp. 201–223. [CrossRef]
23. Koide, K.; Miura, J. Collision Risk Assessment via Awareness Estimation Toward Robotic Attendant. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 11011–11016. [CrossRef]
24. Torrejon, A.; Zapata, N.; Núñez, P.; Bonilla, L.; Bustos, P. Shadow, an accompanying tool robot. In Proceedings of the Workshop on Autonomous Systems 2023, Aranjuez, Madrid, Spain, 9–10 November 2023.
25. Sanz, R.; Rodriguez, M.; Aguado, E. D1.3 Theory of Understanding. Available online: <https://coresense.eu> (accessed on 17 August 2024).

26. Bustos García, P.; Manso Argüelles, L.; Bandera, A.; Bandera, J.; García-Varea, I.; Martínez-Gómez, J. The CORTEX cognitive robotics architecture: Use cases. *Cogn. Syst. Res.* **2019**, *55*, 107–123. [[CrossRef](#)]
27. Shapiro, M.; Preguiça, N.; Baquero, C.; Zawirski, M. *A Comprehensive Study of Convergent and Commutative Replicated Data Types*; Research Report RR-7506; INRIA: Le Chesnay-Rocquencourt, France, 2011.
28. García, J.C.; Bachiller, P.; Bustos, P.; Núñez, P. Towards the design of efficient and versatile cognitive robotic architecture based on distributed, low-latency working memory. In Proceedings of the 2022 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Santa Maria da Feira, Portugal, 29–30 April 2022; pp. 9–14. [[CrossRef](#)]
29. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
30. Zhou, H.; Jiang, F.; Si, J.; Lu, H. Joint Multi-Person Body Detection and Orientation Estimation via One Unified Embedding. *arXiv* **2023**, arXiv:2210.15586.
31. Zhang, Y.; Sun, P.; Jiang, Y.; Yu, D.; Weng, F.; Yuan, Z.; Luo, P.; Liu, W.; Wang, X. Bytetrack: Multi-object tracking by associating every detection box. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 1–21.
32. Sasaki, Y. The truth of the F-measure. *Teach Tutor Mater.* **2007**, *1*.
33. Donoso, M.; Collins, A.; Koechlin, E. Human cognition. Foundations of human reasoning in the prefrontal cortex. *Science* **2014**, *344*, 1481–1486. [[CrossRef](#)] [[PubMed](#)]
34. Kahneman, D. *Thinking, Fast and Slow*; Farrar, Straus and Giroux: New York, NY, USA, 2011.
35. van Maris, A.; Zook, N.; Dogramadzi, S.; Studley, M.; Winfield, A.; Caleb-Solly, P. A New Perspective on Robot Ethics through Investigating Human–Robot Interactions with Older Adults. *Appl. Sci.* **2021**, *11*, 10136. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.