

Model of Side-by-Side Walking Without the Robot Knowing the Goal

Deneth Karunarathne^{1,2}  · Yoichi Morales¹ · Takayuki Kanda¹ · Hiroshi Ishiguro^{1,2}

Accepted: 7 November 2017 / Published online: 17 November 2017
© Springer Science+Business Media B.V., part of Springer Nature 2017

Abstract We humans often engage in side-by-side walking even when we do not know where we are going. Replicating this capability in a robot reveals the complications of such daily interactions. We analyzed human–human interactions and found that human pairs sustained a side-by-side walking formation even when one of them (the follower) did not know the destination. When multiple path choices exist, the follower walks slightly behind his partner. We modeled this interaction by assuming that one needs knowledge from the environment, like the locations to which people typically move toward: subgoals. This model enables a robot to switch between two interaction modes; in one mode, it strictly maintains the side-by-side walking formation, and in another it walks slightly behind its partner. We conducted an evaluation experiment in a real shopping arcade and revealed that our model replicates human side-by-side walking better than other simple methods in which the robot simply moves to the side of a person and without the human tendency for choosing the next appropriate subgoals.

Keywords Human–robot interaction · Side-by-side walking · Concept of subgoal

1 Introduction

Side-by-side walking is one of the important social capabilities of people. People often walk in smaller social groups or in pairs and often communicate with each other while walking [14, 26]. To facilitate communication, people try to maintain personal distance and eye contact. To do so, walking in a side-by-side formation is indispensable. Therefore, side-by-side walking is an important capability of a social robot. There are many scenarios in which social robots will need to talk with people while walking. In the near future, social robots will be used as companions and assistants at various locations, such as shopping malls, museums, hospitals, and elderly care centers. For these social robots, one of the typical situations is to be together with a user who walks around or goes towards some destination. As many people do, it is expected that such a robot engages in a conversation while the person takes the lead to go somewhere; hence, it is important for a social robot to keep the side-by-side formation.

Nevertheless, it is difficult to achieve the capability of side-by-side walking in robots. When performing side-by-side walking, people walk as if they know where they are going, although they may not. Sometimes only one person knows the destination and does not necessarily communicate it to the other person. For example, when a person makes a restaurant reservation and accompanies another person, he/she might only mention the restaurant but not explain in detail how to get there; yet, they would walk towards it together. If two people are window-shopping, most of the time they do not have a particular destination; one might lead (temporarily deciding the goal) while another would follow (not knowing the goal). Even if both partners would know the destination, a similar situation would occur. For example, like illustrated in Fig. 1, when there are multiple routes toward the destination, one would take an initiative while another would follow.

✉ Deneth Karunarathne
deneth9@gmail.com

¹ Intelligent Robotics and Communication Laboratories,
Advanced Telecommunications Research Institute
International, Kyoto, Japan

² Graduate school of Engineering Science, Osaka University,
Toyonaka-shi, Osaka, Japan

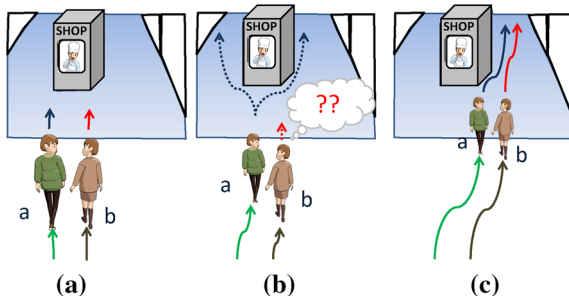


Fig. 1 People walking side-by-side facing an intersection. **a** Going straight, **b** leader guiding, **c** inferring routes

Implementing a velocity based side by side walking robot is straight forward if people move in constant velocity all the time. However, this is usually not the case as human motion has slight changes. Moreover, partner anticipation to compute robot's motion is crucial to achieve collaborative motion.

Likewise, in many situations, when the route to the destination is unknown, the person who knows usually leads while the other person follows. Interestingly such a leader–follower relationship is not very visible from their behavior. Seemingly, they can walk together in side-by-side formation. It is unusual to see two persons walking in a single line (i.e. one following another from behind) unless the environment is crowded. The follower can keep up probably because he/she has an understanding of the environment, previous experience, and is anticipating the direction of their joint motion.

In a previous study, we were only able to replicate the side-by-side walking capability in the case where the destination was known to the robot [15, 16]. In contrast, in this paper, we address the situation where the final destination is unknown for the robot while it tries to engage in side-by-side walking.¹ This is achieved by introducing environmental knowledge about locations that pedestrians might head towards, here after called “subgoals”.

2 Related Work

2.1 Model of Human Walking Behavior

Observations on humans operating as social groups have identified the concept of personal space. Personal space is defined as the space where a person is comfortable while

¹ This paper is an extended version of our preliminary work reported in [17]. The two states model was introduced in this preliminary work. However, while the work in [17] was conducted in a simple indoor corridor with one simple T-shape intersection, this work provides an evidence about the effectiveness of the model in a real shopping mall arcade. For that purpose, we created an algorithm for managing the transition from one state to another based on the estimate of probable subgoals. In addition, the implementation of the robot was fully updated for the use in the real environment.

engaging in an interaction. Hall suggested that this concept of personal space depends on the context [14]. Kendon studied the spatial formation during conversation [11]. In addition, studies in social science have suggested that humans tend to walk in social groups if the opportunity is presented [14, 25]. Costa provided proof that people tend to walk in side-by-side formation unless the environment is overly crowded [2]. The study further discusses that when there are three or more people they start to form more complex formations, such as “V” shapes and side-by-side formations.

Further, in physics and transportation research, researchers have built mathematical models to illustrate human walking behavior. A social force model was first proposed by Helbing and Molnar in order to model pedestrian dynamics [6, 7]. The recent studies addressed to model social group behaviors, in which attraction forces were constantly applied towards either group's center of gravity [14] or the positions of other members [24].

Side-by-side walking was not modeled in these previous studies, and these mathematical models cannot be directly applied for the side-by-side walking modelling. Walking side-by-side is a collaborative task where each agent considers partner's utility in its planning, for which environmental factors are also influential like the locations of subgoals.

2.2 HRI Studies for Locomotive Interaction

A large number of studies have addressed proxemics in HRI (Human–Robot Interaction). That is, they investigated how people keep social distances with robots, and how robots should control social distance with people while they interact (e.g. [9, 23]). Informed by the concept of spatial formation during conversations modeled in human science [11, 13] successfully controlled spatial formation when people talked with a robot.

There are a few studies that addressed the situation when people walked with a robot. For example, a study conducted by Gockley, Forlizzi, and Simmons [5] provided a model for a robot to follow a person from behind. A simulation study addressed the formation for shepherding a group of people [4]. A path-planning algorithm where human comfort level was considered was proposed by [21]. Satake and his colleagues in [20] developed and tested in a real environment a robot approaching behavior based on the anticipated motion of a person.

While in the above studies the robots either led or followed people, side-by-side walking involves a problem of a more complex nature, collaboration. It is not like one entity completely takes the lead while another follows. Due to this collaborative nature, it is difficult to plan a robot's motion. Thus, we consider that our work is original in addressing collaboration in locomotive interaction.

2.3 Side by Side Walking

Until recently, only a limited amount of studies have been conducted on side-by-side walking. To keep a robot alongside a walking person, it is essential to predict the near future location of the person. Without any prediction, if a robot tries to go to the side of the person at a certain moment the walking person will have already moved forward when the robot reaches that location, thus the robot will fail, resulting in being always behind the person. Instead, in the literature the most common idea was to use recently-observed velocity of the person and extrapolate it to predict the future location [12, 19]. However, during side-by-side walking, people often change the direction of their motion. One reason is that they try to be cooperative. They would adjust distance from a robot and adjust their speed to the robot. Another reason is due to the environmental characteristics. They would need to avoid obstacles or change the course of motion to move toward their destination. The velocity-based approach tends to suffer from this unstable tendency of the velocity vector.

Without fully depending on the current velocity of the partner, Morales et al. [15] built a model in which they highlight the importance of joint planning where both the robot and human agents anticipate each other. However, this model has a shortcoming since joint planning requires knowledge of their destination. This means the robot needs to know the final destination and the route that they will take. The work in [16] implemented the side by side walking model in a shopping environment which mainly consisted of straight segments and required both human and the robot to know the destination. In previous work in [17] we tested a two-state model in an indoor laboratory environment with one junction and four subgoals placed manually. The robot needed to anticipate the movement only when it reached the junction and we used a simple distance and angle based method to decide the next subgoal. Such a simple approach does not scale well for larger complex real environments. Compared to the previous works, the study presented in this work presents a side by side walking model that does not require the robot to know the destination in advance. The study is held in a complex shopping mall scenario where multiple subgoals are available at a time. Additionally, a subgoal selection to estimate the target subgoal from possible subgoals is proposed.

3 Analysis of Human Side-by-Side Walking in a Shopping Arcade

3.1 Setup

In our preliminary study reported in [17], we made three observations:



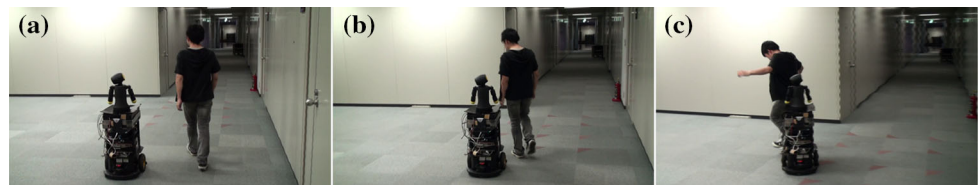
Fig. 2 Map of the environment with snapshots

- Even when one walking partner (the follower) is not informed about the route/destination, they do not explicitly communicate the route/destination.
- People sustain side-by-side formation.
- The follower exhibits an “opening-space behavior” to make it easier for the leader to turn to any direction, around the location where there are multiple routes, i.e. junction.

The above observation was made in a simple office corridor. Thus, it was quite easy to predict where they might go, as the choice can be something like go straight, or turn left.

To confirm whether the above observation is still applicable in a more complex environment, e.g. a shopping arcade, we conducted an additional data collection in a shopping mall arcade with the availability of several shops and guidance announcements (see Fig. 2 for reference). We invited two pairs of participants, who are members of our laboratory but not informed about the study’s hypothesis, to freely walk around for window-shopping. We carefully chose them according to similar social status. Within this context, we setup the situation where one of them knew the destination (leader) and the other one did not (follower). Every participant walked 5 times as leader and 5 times as follower. We collected 10 trajectories per pair using the sensing infrastructure reported in [1], totaling 20 trajectories. In addition, we used this ‘trajectory dataset’ to calibrate parameters of the proposed model.

Fig. 3 Walking together at an intersection where the robot did not open space for human to turn. **a** Walking straight. **b** Leader turns. **c** Robot does not open space



3.2 Observations

We made the following observations in real shopping mall environment that support the evidence reported in the indoor constrained environment [17].

We found that walking partners maintained a side-by-side formation, although we did not tell the followers about the destination.

In [17], we found the importance of opening-space behavior. For instance, Fig. 3 shows one of the scenes in which the person walking on the right is the leader and the robot on his left is the follower. Figure 3 is a scene from [17] where a person interacting with a robot that does not perform opening-space behavior. Around a corner, as the robot mindlessly tries to stay beside to the person, which prevented the person from turning.

In contrary to this example, people behave in a more intelligent way. When walking side-by-side, opening space behavior is important. This is illustrated in Fig. 4. The crucial moment was captured when they turned left at $t = 7.95$. Before they turned, the follower did the opening-space behavior. At time $t = 6.05$, we can observe that the leader was walking straight while the follower, who did not know whether the leader would turn, walked slightly behind the leader, and opened the space on the left of the leader. It means that the follower has already made space for the leader to turn left. If the follower just mindlessly kept walking straight in a strictly side-by-side formation, it would be very difficult for the leader to turn left; if he tries to turn, they would collide or push each other, like the human–robot case shown above.

Figure 5 shows the trajectories of a pair of people in another situation in a shopping mall, which shows the switching between two states. In Fig. 5, at times $5.45 \leq t \leq 21.3$ and $t \geq 50.0$ the follower could easily infer where they were heading along the corridor and stayed directly beside the leader in side-by-side formation. Thus, we named this situation a *collaborative state*. In a collaborative state, their relative angle (angle towards the *follower* from the *leader's* motion direction) was very close to 90° (89.56° , $s.d. = 21.73$). The higher standard deviation can be interpreted as that the follower does not need to observe the leader since the motion direction is known. Thus, allows a degree of freedom within the motion while conducting side by side motion allowing a larger standard deviation. Even though the desti-

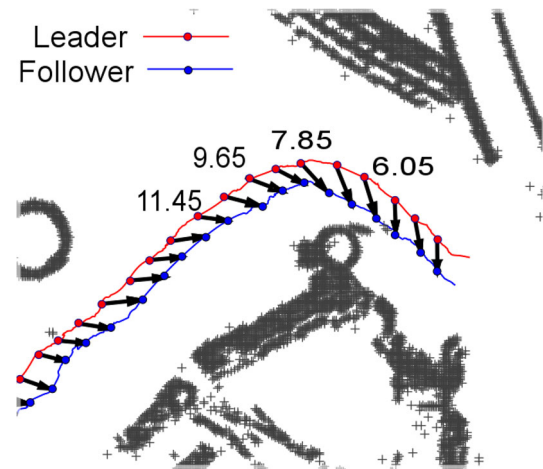


Fig. 4 Side by side walking trajectories from two people in a shopping mall arcade. The follower opens space for the leader when they turn left. In Figs. 3 and 4, timestamp points are shown in bold face numbers, indicating people's relative positions at different points in time

nation was not given, the current path during these moments was apparent to the follower.

In contrast, the motion between the time $28.7 \leq t \leq 48.5$ shows a different pattern. The leader (solid line) kept walking toward the goal and the follower (dotted line) walked slightly behind. In this period of time, there were many possible path choices, turning right for the stairs, turning left to the hallway or continuing straight (see top right of Fig. 5). As discussed above, Fig. 4 also captured such a moment. In average in our shopping arcade data, this approximately 16° of offset of relative angle from 90° (toward the *follower* from the *leader's* motion direction, in all trajectories, avg. 106.59° , $s.d. = 5.90$) has two effects: It enables the follower to observe the leader's movement direction, and it also enables the leader to easily turn without obstructing the follower's walking. Standard deviation of 5.90 observed in the leader follower mode is smaller than the standard deviation of collaborative mode. This is due to the fact that once in leader follower mode the follower observes the leaders motion while keeping distance if the leader makes a turn. Therefore, the follower was always slightly behind to the follower, and has less freedom of motion thus the standard deviation is 5.90. We named this state the *leader–follower state*.

In both states, even in the collaborative state where the follower is quite certain that they are heading in a certain direction, there could be the possibility that the leader would

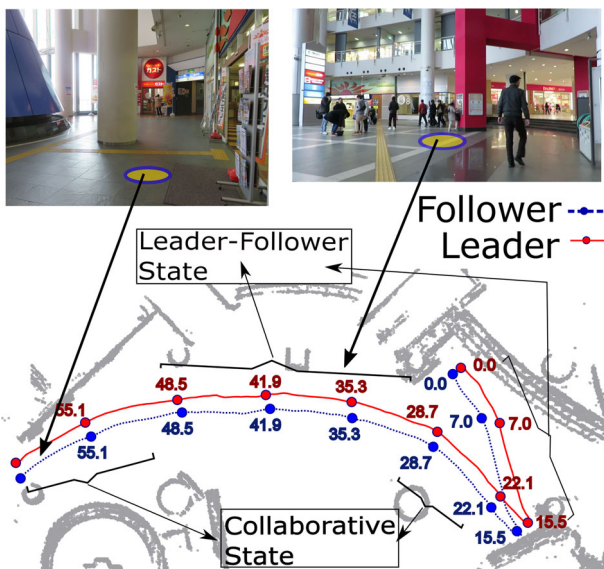


Fig. 5 Side-by-side walking trajectories where two states switch. When there are no choices of routes (snapshot at the top left), they walk in parallel in the “collaborative state”; when there is more than one direction (top right), the follower walked slightly behind in the “leader–follower state”

suddenly deviate and take an unexpected detour e.g., they could suddenly turn toward a signboard or a shop entrance. For instance, in Fig. 5 at time $t \geq 50.0$ (picture is shown on the upper left) it is highly likely that they will keep walking along the current path going between the wall on the right side and the column on the left; however, they could still unexpectedly deviate. In this case they keep collaboratively walking side-by-side (which is different from the leader–follower state), although there can be other less likely but possible deviations. Thus, the follower prepares for such deviations according to how likely he believes it is that the leader will detour from the current path.

In summary:

1. There was no explicit communication of the destination.
2. Partners walked collaboratively in side-by-side formation when both headed toward a defined destination.
3. When the walking partners faced a point where a decision was needed (e.g. possible detour is expected), the person without knowledge of the goal (follower) created space staying slightly behind anticipating possible direction change. In this case, the partner with knowledge of the destination (leader) walked toward the goal.
4. We observed that the follower partner estimated the most likely path and the possible detours from their current location and trajectory. However, the follower did not consider possible but less likely paths (such as would be a path that is was behind and in the opposite direction from their trajectory). Thus, a computation of “likely” path is key for the follower.

Observations 1, 2 and 3 confirmed the evidence presented in [17]. Additionally, observation 4 was evident as a result of experimentation in real environments with multiple possible detours.

4 Side by Side Walking Modelling

4.1 Overview

From the observations section, we found that it is vital to replicate whether a follower would believe that there are multiple possible directions or not. For this, our assumption is that people would have similar perceptions, so how a person would behave (and, whether there are multiple possible directions to be perceived) could be inferred from trajectories of many other people. We will explain how we retrieve the possible directions, represented as “subgoals”, from trajectories, and how we estimate current probable subgoals in the subgoals section.

The estimation result is used to determine the transitions between two states, collaborative state and leader–follower state. In the two-state model sections, we will discuss the details of the two state model. We will also provide details of the mathematical modelling for each state. Finally, the parameters for model are calibrated with the dataset of human pairs walking in side-by-side.

4.2 Subgoals

4.2.1 Concept of Subgoal

Subgoals are points and landmarks in the environment toward which pedestrians walk or make directional choices about before reaching their destination [10]. We assume that people walk toward a subgoal but not necessarily through each subgoal; instead, we assume that the transition between subgoals occurs prior to their arrival. It is represented with a transition radius ($r_{1 \rightarrow 2}$). The transition radius from one subgoal (s_1) to another subgoal (s_2) is calculated by the distance when the pedestrian changed the direction from current subgoal s_1 to the next subgoal s_2 .

4.2.2 Technique to Retrieve Subgoal Locations

We used a technique proposed by I. Ikeda and Y. Chigodo in 2012 [10] to compute subgoal locations. The trajectories collected in the environment are solely responsible for the determination of the subgoals. The basic idea is to analyze pedestrians’ walking trajectories, and retrieve the location that their velocity vectors often head toward. To compute these directions of velocities from the trajectories are accumulated at every 0.5 m grid of the environment, and then a

couple of peaks of velocity directions at each grid are statistically estimated from the distribution of velocity directions. Finally, subgoals are retrieved as the points where a larger number of peak directions converge. More details can be found in [10].

The transition between subgoals s_1 and s_2 was probabilistically computed as $p(s_1 \rightarrow s_2) = \frac{\#trajectories\ to\ s_1\ then\ s_2}{\#trajectories\ to\ s_1}$. The transition radius ($r_{1 \rightarrow 2}$) is computed as the average distance when a person currently at s_1 heads towards subgoal s_2 .

For our study, the human trajectories were detected and tracked with the on-board human tracker system. The subgoals were computed off-line with the subgoal extraction method of [10]. We computed subgoals and transition probabilities from 6088 trajectories. The obtained subgoals are shown later together with the experiment result in Fig. 14.

4.2.3 Subgoal Estimation

We represent the follower agent’s belief about the current subgoals as follows:

Possible subgoals (S_P) are the set of subgoals that could be followed next.

Target subgoal (s_{target}) is the subgoal towards which he/she is moving now.

It is important to estimate S_P , because it decides whether a follower agent would be with leader–follower state or collaborative state.

In our preliminary work [17], we estimated S_P based on distance and visibility, i.e. all the neighbor subgoals within the follower’s sight were considered as S_P . However, in a real complex environment, there are often multiple direction choices, i.e., there are several subgoals within one’s field of view most of the time. Such availability of multiple subgoals is one of the differences from our preliminary work [17] where the number of subgoals in an indoor corridor.

For this problem, we created a new algorithm. Our idea is that people have similar tendencies in understanding environments. Hence, one would likely turn at a location where many other people turned, and one would be less likely to turn at a location where only few others did. Thus, we added a transition probability $p(s_1 \rightarrow s_2)$ into the estimation algorithm in addition to visibility.

We calculate possible subgoals S_P based on the partner’s current position p_t^j and orientation. First, we create a visual cone C (Fig. 6) with a 180° field of view. All n subgoals inside the cone are considered ($S_{visible}$). From current subgoal $s_{current}$, the subgoals in $S_{visible}$ with small transition probability value t_h are removed. We used the following equation to evaluate whether the next subgoal s_{next} is possible from current subgoal $s_{current}$:

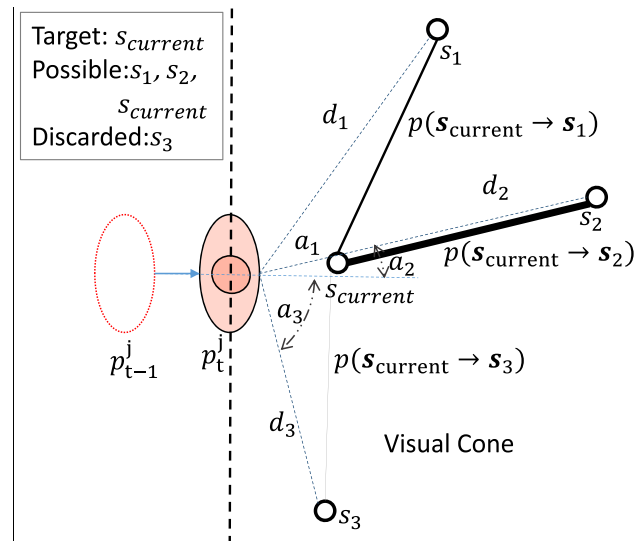


Fig. 6 Subgoal selection algorithm. Possible subgoals are selected from visual cone based on angle, distance, and transition probability

$$f(s_{current}, s_{next}, p_t^j) = p(s_{current} \rightarrow s_{next}) \left(1 - \frac{|s_{current} - p_t^j|}{r_{current \rightarrow next}} \right) \quad (1)$$

where $p(s_{current} \rightarrow s_{next})$ is the transition probability between current subgoals $s_{current}$ and s_{next} . $r_{current \rightarrow next}$ is the transition radius.

Equation 1 provides all the visible subgoals within the visible cone after removing the subgoals that does not have a sufficient transition probability. Thus, removing any subgoal that is within the visual cone but actually un-attainable. Lack of transition probability via a certain subgoal is often associated with that subgoal been visited poorly through the current subgoal.

If $f(s_{current}, s_{next}, p_t^j) > t_h$ then subgoal s_{next} is a possible subgoal S_P . We empirically set the threshold to $t_h = 0.05$ which appropriately discard subgoals that are not perceived as a possible deviation and hence reproduced the transition between two states. In our experiment setting, this value resulted in discarding the lowest 30% of subgoal transition probabilities.

From the set of possible subgoals S_P we selected the one in frontal orientation and near the human partner as the target s_{target} . s_{target} is calculated by the distance from person to the subgoal and the angle from person to the subgoal. It is known that humans prefer not to change the direction. The closer it is, and less orientation change that has to be made is used in calculating the target subgoal. That is, for each subgoal k in S_P , we calculated the angle (a_m) and distance (d_m) from the partner’s point of view. The angle is computed as $a_m =$

angle $\left(p_{t-1}^j, p_t^j, p_t^j s_k\right)$ where $\text{angle}(\vec{p}, \vec{q})$ is a function that returns the relative angle of vectors \vec{p} and \vec{q} . Distance is computed as $d_m = \text{distance}\left(p_t^j, s_k\right)$ where $\text{distance}(p, q)$ is a function that returns the distance from point p to q .

Finally we used the following equation to compute the target subgoal s_{target} :

$$s_{target} = \text{argmax}_{s_k \rightarrow S_P} N(a_m) N(d_m) \tag{2}$$

where $N(x_x) = \frac{1}{\sigma_x \sqrt{2\pi}} e^{-\frac{x^2}{2\sigma_x^2}}$ is a centered Gaussian function with standard deviation σ_x . The target subgoal is the subgoal with the highest peak after multiplication. The Eq. (2) considers the two factors in calculating the target subgoal which are distance and angle denoted by d_m and a_m respectively. The standard deviation for each faction is set observing the environment. The equation provides the balance between distance and angle when selecting a matching subgoal rather than overly depending on one factor.

We empirically set values are $\sigma_{a_m} = 22.5^\circ$ and $\sigma_{d_m} = 20$ m. The angle value was selected in order to consider subgoals in front of the human. Distance value was set to be a large value, which resulted in adding a preference to a nearby subgoal rather than far ones in case their angles are similar.

4.3 Two-State Model

To replicate the tendency of walking pairs presented in the observations section, we propose a two state model.

Without the follower being not explicitly informed where they go, when he can infer where they go, they walk in alongside, i.e. their relative angle is very close to 90° . This is referred as “collaborative state”.

Meanwhile, when there are multiple subgoals to choose from and the follower does not know which one to follow, he lets the leader show the way walking in a leader-follower state. He is slightly behind the leader. In such a moment, the follower is unsure where they are going, and it is too risky to blindly choose one direction from many possibilities. Instead, the slightly behind location enables him to be flexible for any choice the leader will make.

The transition between states happens with the certainty that the follower has of the next subgoal. Figure 7 shows the two state model and the state transition.

The walking state is decided on the set of possible subgoals S_P as shown in the following equation:

$$\text{State}(S_P) = \begin{cases} \text{Collaborative,} & \text{if } |S_P| = 1 \\ \text{Leader-Follower,} & \text{if } |S_P| > 1 \end{cases} \tag{3}$$

where $|S_P|$ is the size of the set. In the case that there are no possible subgoals $|S_P| = 0$, robot operates in collaborative state without subgoal utility.

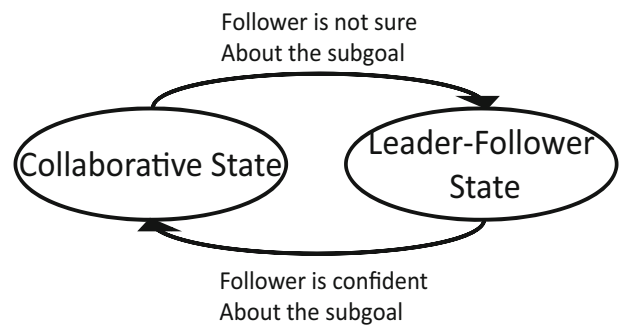


Fig. 7 Two state walking model composed of collaborative and leader-follower states

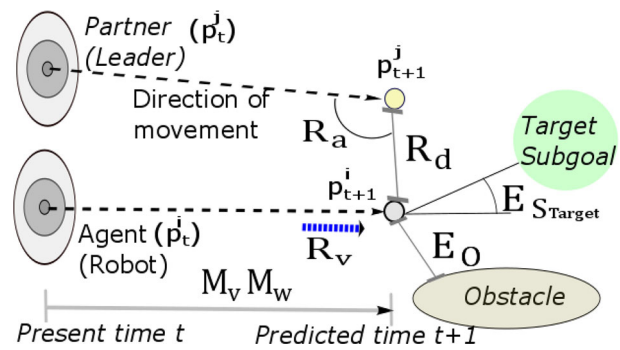


Fig. 8 Utility function’s factors

4.3.1 Collaborative State

Walking partners are in the collaborative state when the follower can infer the targeted subgoal with certainty. We can model this situation to be equivalent to the situation in previous work [15] where both agents know the following subgoal, while both agents try to act collaboratively to maintain their relative positions. Thus, for this state, we applied our previous model.

We present a brief introduction to previous work; more details are available in the earlier paper [15]. The model has two important concepts, joint planning and representation of the goodness of future motion as a utility function.

Walking partners jointly plan their future motions. They anticipate their partner’s future utility motion (Eq. (6)) and maximize joint utility (Eq. (7)). For the utility function, we combined eight different factors. Figure 8 illustrates all eight factors. From the perspective of the ‘agent’, its utility is high if it moves toward the subgoal (represented by subgoal utility E_S), if it is far from obstacles (obstacle utility E_O), if its relative position is precisely alongside the partner (relative utilities: R_a , R_d , and R_v), and if its motion is sufficiently smooth (motion utilities: M_w , and M_v). Given the agent’s position at time t denoted as p_t^i , each utility is defined as follows:

Obstacle utility (E_O) This is modeled as the following function, which is often used to represent the potential field for obstacles:

$$f_o(x) = - \left| \left(\frac{x}{a} \right)^{-2b} \right| \tag{4}$$

E_O is defined as $f_o(d_{closest}(p_{t+1}^i))$, where $d_{closest}(p)$ returns the distance to the closest obstacle at position p . The utility will be higher if the distance to the obstacle is farther.

All other utility functions are modeled as power functions (Eq. (5)) because their distribution has a peak and is roughly symmetrical.

$$f(x) = - \left| \left(\frac{x-c}{a} \right)^{2b} \right| \tag{5}$$

This power function has three parameters. Parameter “ a ” is related to the tolerance of the function, parameter “ b ” corresponds to the curve steepness factor, and parameter “ c ” is the curve’s center where the function yields the maximum value.

Subgoal utility ($E_S(s_{target})$) This computes the utility for moving toward the target subgoal s_{target} , which is defined as $f(\text{angle}(\vec{p}_{t+1}^i, \vec{S}p_{t+1}^i))$. A better utility is provided if the motion vector heads towards the subgoal.

Motion utilities These represent the smoothness of the motion. Given that velocity $v_{t+1}^i = (p_{t+1}^i - p_t^i) / \Delta t$, two utilities are defined as preferred velocity utility $M_v = f(v_{t+1}^i)$ and angular velocity utility $M_w = f((\theta_{t+1}^i - \theta_t^i) / \Delta t)$. These utilities are high if the agent moves straight at a constant speed.

Relative utilities These represent the goodness of position relative to the partner. They include relative distance $R_d = f(|p_{t+1}^i - p_{t+1}^j|)$, relative angle $R_a = f(\text{angle}(\vec{p}_{t+1}^i, \vec{p}_{t+1}^j))$ and relative velocity $R_v = f(v_{t+1}^i - v_{t+1}^j)$. The highest utilities are obtained if the partners are strictly alongside at the same velocity.

The overall utility U_C for agent i positioned at point p_{t+1}^i toward its partner’s position p_{t+1}^j and moving toward a subgoal S_E is given as:

$$U_C(p_{t+1}^i, p_{t+1}^j, s_{target}) = k_{E_O} E_O + k_{E_S} E_S(s_{target}) + k_{R_d} R_d + k_{R_a} R_a + k_{R_v} R_v + k_{M_v} M_v + k_{M_w} M_w \tag{6}$$

where coefficient k_x is the parameter that represent the weight of utility x (e.g. E_O, R_d, \dots). The next desired position of agent i walking with partner j and toward a targeted subgoal

s_{Target} is chosen to maximize the utility calculated by the following expression:

$$\hat{p}_{t+1}^i = \text{argmax}_{\{p_{t+1}^i | P^i\}, \{p_{t+1}^j | P^j\}} \left\{ U_C(p_{t+1}^i, p_{t+1}^j, s_{target}) + U_C(p_{t+1}^j, p_{t+1}^i, s_{target}) \right\} \tag{7}$$

where P^i and P^j are sets of neighboring locations each agent could move to. This equation represents the agent maximizing its own utility as well as that of its partner. In short, the planning chooses the next location to be the best for both agents.

4.4 Leader–Follower State

In the observations section, we described that the follower slowed down and walked slightly behind the leader, while the leader (who knows the subgoal) continued walking without changing his behavior. The leader–follower state models this situation.

Model of leader Although our primary focus is to reproduce the *follower* agent in a robot, we need a model for the *leader* to compute the utility for its partner. Since the *leader* knows the destination and decides the subgoals to which they are heading, the *leader* in this leader–follower state can be in an equivalent situation with the model in collaborative state. The question here is whether the leader’s behavior can be represented with the same model in the collaborative state.

We analyzed the trajectories obtained in the data collection to investigate whether people’s behavior in two states are similar. We compared their velocities in two states. We analyzed the trajectories obtained in the data collection and compared the velocities of the leader agents. The average velocity computed from all trajectories was 0.958 m/s (s.d. 0.32) during *collaborative* state and 0.993 m/s (s.d. 0.33) during *leader–follower* state. This means that the leader agent walks at a constant pace regardless of the state. We did not find any additional evidence that would suggest that leader had to be modeled in a specific manner, therefore, we approximated leader’s behavior to be equal between the two states (*collaborative-state* and *leader–follower state*).

Accordingly, the leader agent’s planning is based on Eq. (6). In the equation, it is assumed that both partners know the subgoal (in reality the follower does not); however, in the *leader–follower* state, the leader behaves in the same way as in the case where the partner knows the subgoal. This assumption implies that the leader does not pay special attention to whether the follower knows the subgoal or not.

Model of the follower The *follower* may not be able to infer which subgoal to choose when there is a choice of routes. Until the follower becomes sure about the next subgoal, the

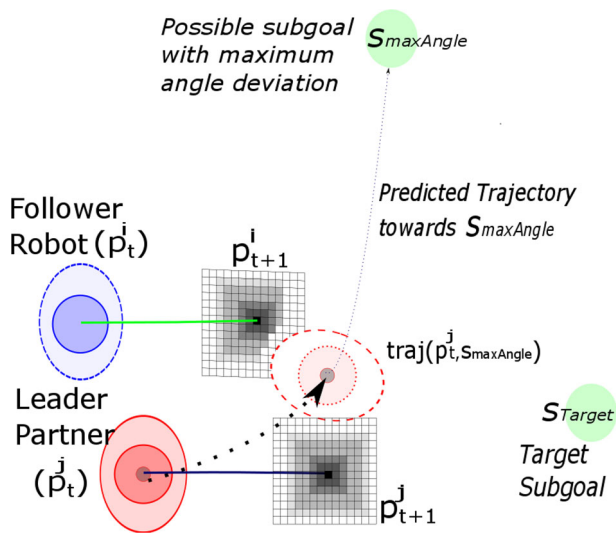


Fig. 9 Utility computation for follower agent in leader–follower mode. Opening-space utility is computed for the subgoal with the largest angular distance $s_{\max\text{Angle}}$

state stays in this *leader–follower state*. In such a situation, an important feature of the follower’s motion is *opening-space* behavior (as reported in observation section). The follower moves in a way to open space so as not to obstruct the leader’s possible turning motion.

We modeled this *opening-space* behavior using the concept of a subgoal. Figure 9 illustrates a situation where opening-space behavior is needed. While the leader and follower walk together, it is expected that the leader goes straight toward target subgoal s_{target} but there is also the possibility that he might turn toward $s_{\max\text{Angle}}$. Without opening space behavior, a follower could block the leader’s path resulting in possible dangerous situations. Thus, the follower needs to stay slightly behind in order to allow the leader to walk freely toward any of the possible subgoals ($S_P \ni s_1, \dots, s_n$). For computational simplicity, we only considered the subgoal with the largest angular distance from the current position (p_t^j) that could intersect with his own future trajectory (same side as follower) ($s_{\max\text{Angle}} = \text{argmax}_{S_P} \{\text{angle}(p_t^j, s_i)\}$). If the follower is able to open space for possible motion towards $s_{\max\text{Angle}}$, the follower’s motion will not hinder the leader’s motion toward any other possible subgoal.

We modeled the above opening-space behavior by adding a new utility named the *opening-space utility* (E_{OS}). For each predicted location p_{t+1}^i this is denoted as:

$$E_{OS} = f_o \left(\min_{\{p \in \text{traj}(p_t^j, s_{\max\text{Angle}})\}} \left(|p_{t+1}^i - p| \right) \right) \quad (8)$$

where $\text{traj}(p_t^j, s_{\max\text{Angle}})$ is the predicted trajectory of the leader moving toward subgoal $s_{\max\text{Angle}}$ composed of a set of locations $p_t^j, \dots, p_{t+1}^j, \dots, p_{t+\Delta t}^j$ where $\Delta t = 2s$. E_{OS} is

computed as a function from the minimum distance from p_{t+1}^i to any point p in the trajectory. In addition, the follower plans his future motion towards target subgoal (s_{target}) for the computation of subgoal utility (E_S) keeping away from the leader’s possible trajectory toward $s_{\max\text{Angle}}$. This results in the follower opening space in the case where the leader turns with the largest deviation toward subgoal $s_{\max\text{Angle}}$. Overall, the utility can be computed with the addition of opening space behavior as follower allows the leader to safely and freely show the way to the next subgoal.

In the open space behavior, the utility E_{OS} effectively allows the robot to be pushed back when there is a possible subgoal in the robot’s side, such that robot would not collide with the human given human changes its direction towards that subgoal. However even in leader follower mode the E_{OS} can be useless if there is no subgoal with a large angle in the side of the follower. In such cases E_{OS} becomes zero (as $f_o(x)$ goes to zero) a large negative value for all the grids. In such cases the E_{OS} could be ignored while calculating the overall utility.

Based on this, the follower’s overall utility U_F is computed as:

$$\begin{aligned} U_F \left(p_{t+1}^i, p_{t+1}^j, s_{\text{target}} \right) &= k_{E_{OS}} E_{OS} + k_{E_O} E_O \\ &+ k_{E_S} E_S \left(s_{\text{target}} \right) + k_{R_d} R_d + k_{R_a} R_a \\ &+ k_{R_v} R_v + k_{M_v} M_v + k_{M_w} M_w \end{aligned} \quad (9)$$

Finally, the next location of the follower considering the opening-space behavior utility U_F is computed by jointly planning with the leader when heading toward the target subgoal s_{target} maximizing the following equation:

$$\begin{aligned} \hat{p}_{t+1}^i &= \text{argmax}_{\{p_{t+1}^i | p^i\}, \{p_{t+1}^j | p^j\}} \\ &\left\{ U_F \left(p_{t+1}^i, p_{t+1}^j, s_{\text{target}} \right) \right. \\ &\left. + U_C \left(p_{t+1}^j, p_{t+1}^i, s_{\text{target}} \right) \right\} \end{aligned} \quad (10)$$

This equation expresses that the follower agent also maximizes its own utility as well as that of the leader, in a similar way to that defined in Eq. 7; however, the difference is that the follower agent’s utility is replaced with U_F . In short, the follower makes additional effort to create space for the leader.

4.5 Parameter Calibration

To provide stable side-by-side walking motion, we calibrated the parameters for utilities in Eq. (6) for the collaborative state and Eq. (9) for the follower state. We used the same side-by-side human walking trajectory data set used for base parameter identification of observation section (20 trajectory-

ries). We fed the trajectory data to a simulator with trajectory data where each utility parameter set were calibrated one by one taking the following steps:

1. *Setting the initial parameters* The utility function which used to calculate the utility of each parameter is denoted as Eq. (5). Initial parameters (a) and (c) for each utility for the Eq. (5) is set to match the mean (c) and standard deviation of the distribution (a) of the data set. Initial values of constants b and the coefficient k_x are set to 1 to minimize the initial search space.

2. *Calibration of parameters for each utility function* Obstacle utility has 2 parameters (Eq. 4). All the other utilities as shown in Eq. (5) has 3 parameters. For each parameter we predefine an interval and a range with minimum and maximum values based on the capabilities of the robot and the recorded human–human side-by-side walking trajectories. Then the following steps were followed.

- a. First, we selected one of the utilities (e.g. relative distance) to calibrate while keeping the other utilities unchanged with the values set in the step 1.
- b. In the simulator, for every time step t , the current positions of follower and leader agents (p_t^i, p_t^j) are read from recorded trajectories. The next possible grid positions for the follower agent at (\hat{p}_{t+1}^i) is then predicted (using equations 7 or 10, based on the state). The grid pair ((\hat{p}_{t+1}^i) and (p_{t+1}^j)) with highest utility is then selected. The prediction error ($e = \hat{p}_{t+1}^i - p_{t+1}^j$) is computed comparing the predicted position with the recorded trajectory of the follower agent. The prediction error was averaged for each complete trajectory. This step is repeated for all recorded trajectories. By adding up the average prediction error for each trajectory, we can calculate the total error for a given parameter set on all recorded trajectories.
- c. In the next step, a different set of utility parameters were used while the rest were unchanged. The new set of parameters were calculated using grid search within predefined range and interval for the selected utility (e.g. relative distance parameters **a**, **b** and **c** are now changed). Step **b** is repeated with the new utility parameter set and the average error is calculated.
- d. Once the grid search space for the utility parameters is exhausted the utility parameters which minimizes the error is selected. It is then updated as the initial parameters for the utility (e.g. the initial **a**, **b** and **c** parameters for utility relative distance will now be replaced with the values identified here).
- e. The process is then repeated from step **a** for the next utility (e.g. relative angle).

Table 1 Parameter calibration results

	Collaborative				Leader–Follower			
	<i>a</i>	<i>b</i>	<i>c</i>	<i>k</i>	<i>a</i>	<i>b</i>	<i>c</i>	<i>k</i>
R _d	0.13	2.5	0.80	0.5	0.23	2	0.85	1
R _a	22.3	1.95	84.3	0.5	33.9	1.2	89.1	1
R _v	0.16	3.75	0.05	0.2	0.13	3.5	0.12	0.12
E _O	0.5	7	–	0.2	0.5	4	–	0.5
E _S	34.5	1.75	0	0.2	45.9	2.25	0	0.75
M _v	0.1	0.95	0.9	0.3	0.2	1.95	0.9	0.2
M _w	14.3	2.5	0	0.5	28.7	3.75	0	0.3
E _{OS}	–	–	–	–	0.35	4	–	1

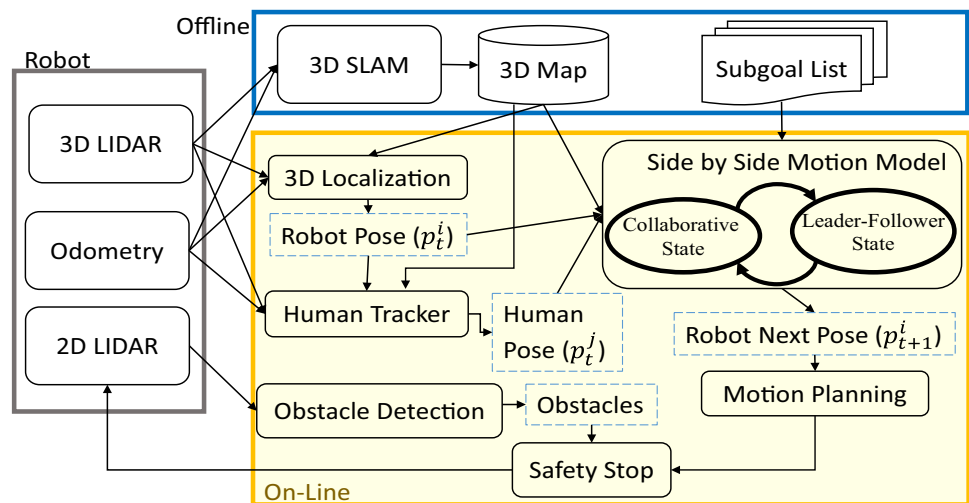
3. *Calibration of coefficient k_x* By this point all the utility parameters are calibrated. The remaining is to calibrate the k_x coefficient. The coefficient k_x denotes the weight of each utility in the utility functions denoted by Eqs. 6 and 9. In a similar process to the step 2 the coefficients changed between predefined range. The resulting parameter set with the minimum error was thus selected as the tuned parameter set.

Final calibrated values are shown in Table 1. The reason for the parameters been different in two states is that the trajectories we see with human–human motion there was some slight difference of the trajectory when the follower was faced with multiple paths as shown in Fig. 5. For both states, c parameter (that corresponds to the peak of utilities) for relative distance (R_d) and relative angle (R_a) stay similar to the average of human data, having relatively high weight (represented by higher k parameters). Both states prefer an angular velocity of 0°/s as their c parameters are zero. The open space utility is only defined for leader follower mode. Yet it has a relatively larger utility than the linear velocity M_v and relative velocity R_v utilities. Both of them have much smaller k values. In the leader–follower state, the follower needs to change his velocity, sometimes to open space and sometimes to catch up, and in occasions go at different velocities than his partner to adjust.

5 System Implementation

5.1 Overview and System Architecture

A robot walking alongside a human partner is required to be fast enough to keep pace and highly reactive to make adjustments with the partner. It also has to localize itself and track the partner in dynamic environments. In addition, several software functionalities must operate simultaneously in real time. Figure 10 depicts the overall architecture of

Fig. 10 System configuration

the system. The system is composed of a fast mobile robot that can compute odometry and is equipped with 3D and 2D laser sensors. In the system, the 3D map and the environment subgoal list need to be prepared beforehand. The rest of the components run on-board in real time.

5.2 Hardware Configuration

We used a Robovie-R3 mobile robot (Fig. 12, left), which is capable of keeping pace with average human walking speed with a maximum speed of 1.20 m/s and maximum acceleration of 0.9 m/s². The robot has a wheeled platform and a human-like physical appearance capable of interacting with people through utterances and gestures. It is 1.10 m tall (1.40 m to the top of 3D sensor) and 0.5 m in diameter.

For map building, localization and human tracking the robot was equipped with a 3D laser sensor HDL-32E Velodyne LIDAR. A 2D Laser sensor (UTM-30LX from Hokuyo) was attached on the bottom to detect obstacles and stop in the case of evident collision.

5.3 3D Map Building and Localization

For robot localization and autonomous motion, a map of the environment is required. To build the map we drove the robot around the environment with a joystick as we logged odometry and 3D range data. We used the 3D Toolkit SLAM library to match consecutive scans and perform global relaxation [18]. We down-sampled the density of the resulting point cloud in octree representation with 0.2 m. of resolution using the octomap library [8]. Using this map as a base, we made a likelihood distance map for localization.

We built a 3D localization module based on a particle filter [3] where each particle corresponds to a possible robot pose $\mathbf{x} = [x, y, z, yaw, pitch, roll]$ and has a weight w_i (x, y and z denote 3D position and $yaw, pitch, roll$ the

orientation). In the prediction step $p(x_t|u_{t-1}, x_{t-1})$, current pose state x_t is computed from previous state x_{t-1} and robot motion u_{t-1} utilizing a differential drive model to propagate the particles. For the update step $p(z_t|x_t, m)$ we used a 3D laser scan $z_t = (z_t^1, \dots, z_t^K)$ and the likelihood field map (m) to compute the weight w_i for each particle using the likelihood end point model [22]. In our implementation, we used 100 particles.

We found that for large shopping environments with many people walking around having a 3D long range localization system was vital to keep the robot pose estimation accurate. Even if people would gather around the robot, with a 3D map and a long range sensor the robot would keep track of its pose. Moreover, shopping mall environments are not totally flat and tend to have slopes.

5.4 Human Tracking

For people detection and tracking, we used the 3D laser scan information. The process is detailed below:

1. *Background subtraction* From the obtained point cloud, we erased points corresponding to walls, as well as points close to walls (in particular the ones far from the sensor) that could create false positives. We used the environmental 3D map m and built an extended map $m_{extended}$ inflating occupied voxels 0.20 m. The difference between point cloud z_t and $m_{extended}$ produces a reduced point cloud $z_{reduced}$ mostly belonging to moving objects.
2. *Clustering* Clustering was done on the reduced point cloud $z_{reduced}$ sorted by height. Starting from the non-clustered highest point, distance (d) toward existing cluster centroids was compared. If smaller than a threshold d_{th} the point is merged to the cluster. In the opposite case, a new cluster is created. The resulting cluster set is possible humans in the environment (Fig. 11 shows exam-

Fig. 11 Result of clustering. (Left: points nearby the robot. Right: points far from the robot, hence points are sparse and larger threshold needed). Sorting points by height allow the clustering of humans close to each other

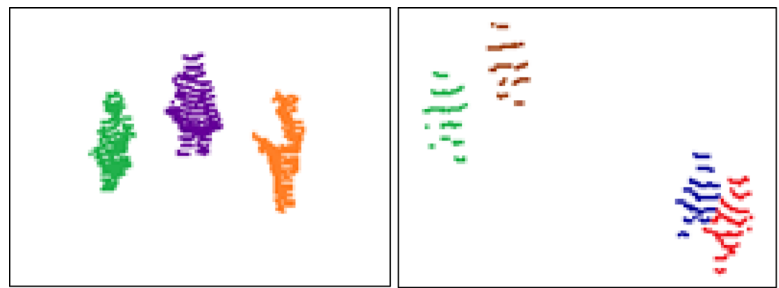
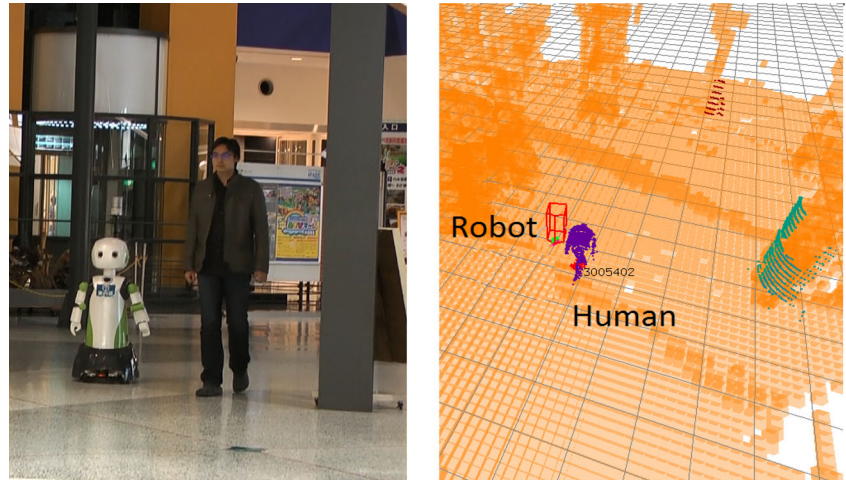


Fig. 12 Result of localization, background subtraction, and human tracking. Extended map is in orange and tracked human in purple



ples). Given the nature of our laser sensor, the threshold value is a function of distance to robot, $d_{th} = f(d_{robot})$.
 3. *Tracking* Computed clusters are fed to a cascade of particle filters. Every tracked entity was assigned a single filter composed of 50 particles per human to estimate 2D positions $\mathbf{x}_{human} = [x, y]$ in global coordinates.

Figure 12 illustrates the result of above computations. The system performed the robot localization, background subtraction, clustering, and human tracking. We did not evaluate the overall performance in accurate way, but it worked quite stable for a single person within a few meters from the robot without occlusion. Hence, during our experiments in the shopping mall arcade, we did not have tracking failures, because the human partner was typically nearby the robot. Regarding the tracking of other humans, in occasions tracking could be lost when there were groups of people walking together and occlusions occurred.

5.5 Side-by-Side Motion Planning

The side-by-side motion planner requires the robot's and partner's current state, then computes future position based on the utility explained in modelling section. Finally, the robot executes the correspondent motion to achieve the high-utility position.

Input parameters for each agent $q_{i,j}$ are current positions (p_t^q), linear and angular velocities (v_t^q, w_t^q respectively). To compute utility of motion, we used sets (P^i, P^j) of future possible neighboring locations for robot and partner. Sets of locations were implemented as “anticipation grids” with cell resolution of 0.2 m . The center of each grid is located on the extrapolation of current linear and angular velocities of each agent $q_{i,j}$ given by $p_{t+1}^q = p_t^q + v_t^q t_{pred}$, where extrapolation time t_{pred} was set to 2 s.

The robot is constrained by its differential-drive wheel configuration and linear and angular velocities and accelerations. Only grids satisfying these constraints were selected. From current position p_t^i , we propagated linear and angular velocities through the robot model where future achievable locations are given by $p_{t+1}^i = p_t^i + \sum_{k=t}^{t_{pred}} (\text{model}(v_k^i, w_k^i) \Delta t)$. Figure 13 illustrates the anticipation grids containing future possible neighboring locations (P^i, P^j) for robot and partner.

6 User Evaluation

6.1 Hypothesis and Prediction

As summarized in related works section, there is mainly only a *velocity-based-prediction* approach that is applicable to our situation, in which a robot is controlled to go

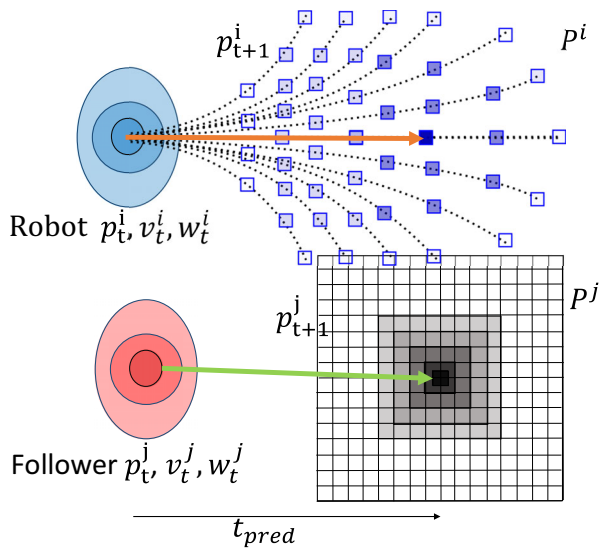


Fig. 13 Anticipation grids to compute future possible position sets for robot P^i , and leader P^j

alongside the projected position of its partner. Previous studies such as the one in [19], proposed a method with linear extrapolation of velocity without proposing a way to jointly anticipate either the robot’s or its partner’s position. The study in [12] also proposes extrapolation of velocity with pedestrian’s body heading extraction with the laser scanner which does not include partner anticipation. We compared the proposed model with this *velocity-based-prediction*.

However, ‘knowing a goal’ is essential. As our previous work [15] revealed, without having appropriate anticipation of the future utility of both oneself and the partner, the interaction is not natural. In this study, by using subgoals we gain knowledge on what might be the local destination. The combination of environmental knowledge, including subgoals as well as obstacles and partner’s motion, enables reliable prediction better than the velocity-based approach. Thus, if our new model is appropriately prepared, it should permit appropriate anticipation of the future utility (even though the destination is not told, which is different from [15]), thus the robot’s motion will be more stable. This should be visible by relative distance and angle the robot keeps with the partner. Thus, we made the following prediction:

Prediction 1 With the proposed model, the robot will provide relative distance and relative angle values closer to the values obtained in human side-by-side walking than the velocity-based method.

The closer these values come to the values observed in human side-by-side walking the more natural and stable the motion perceived by the participants in the experiment. Thus, we made the following prediction.

Prediction 2 A robot with the proposed method will provide a more natural impression and will be perceived as safer, and thus better in its overall evaluation than the model in which subgoals are not used and in which the subgoal selection method is not used.

The proposed model uses subgoals to anticipate the partner’s future motion. If we look at the details of our mechanism, there are two key elements related to subgoals. First, the robot predicts the current likely target (achieved with the ‘subgoal utility’), and second, it is prepared for possible turning behavior (achieved with the ‘opening-space utility’), for which we believe that the ‘subgoal selection method’ (presented in modelling section) is an indispensable mechanism. We believe that this ‘subgoal selection method’ should prevent the follower robot from unnecessarily moving behind the leader when anticipating a change of direction. Based on this idea, we made the following predictions:

Prediction 3 With the proposed model, the robot will provide relative distance and relative angle values closer to the values obtained in human side-by-side walking than the method without subgoal-selection.

Prediction 4 With the proposed model, the robot will provide a more natural impression and will be perceived as safer, and thus better in its overall evaluation than the method without subgoal selection.

6.2 Method

1. *Participants* Participants were 15 Japanese people (9 males and 6 females, average age 24.8 years), who were paid for their participation.

2. *Conditions* We prepared a within-subject experiment with one factor, control method. Three conditions were prepared. The order of the conditions was counter-balanced.

Proposed model The robot navigates with the proposed model (Eq. 10) described in modelling section.

Velocity-based The robot navigated with velocity based prediction. The partner’s future position (\hat{p}_{t+1}^j) is predicted from linear extrapolation of velocity, computed as: $\hat{p}_{t+1}^j = p_t^j + v_t^j \Delta t$ where p_t^j and v_t^j are the observed position and velocity at the time step t . We used a utility-based planning framework similar to what we did for the proposed method. That is, only removing the subgoal utility from Eq. 6, we define overall utility in the velocity-based approach $U_v(p_i, p_j)$ as:

$$U_v(p_{t+1}^i, p_{t+1}^j) = k_o f_o + k_{R_d} f_{R_d} + k_{R_a} f_{R_a} + k_{R_v} f_{R_v} + k_{M_v} f_{M_v} + k_{M_w} f_{M_w} \tag{11}$$

Among all possible positions P , the next position is selected to maximize the above utility in (Eq. 11):

$$\hat{p}_{t+1}^i = \operatorname{argmax}_{\{p_{t+1}^i | P^i\}} U_V \left(p_{t+1}^i, \hat{p}_{t+1}^j \right) \quad (12)$$

All parameters k_x are calibrated to best fit the trajectories collected in data collection in the same way we did for the proposed method.

Without subgoal-selection The robot navigated with the proposed model (Eq. 8), but did not use transition and radius probability information. Instead, we compute visible subgoals $S_{visible}$ by only calculating relative angle (a_n) of the moving robot towards each n subgoal s_n of the whole set N . The angle is calculated by $R_a = f \left(\operatorname{angle} \left(\vec{p}_{t-1}^i p_t^i, \vec{p}_t^i s_n \right) \right)$. All s_n subgoals that satisfy condition $a_n < 90^\circ$ are considered part of the visible subgoal set $S_{visible}$. Algorithm to select S_P from $S_{visible}$ is not utilized.

3. Procedure The experiment was conducted in the same shopping mall environment, for which we retrieved subgoal location as shown in Fig. 14. We asked participants to navigate the environment as they were window-shopping (among shops, ATM, game center, piano etc.) in the environment while the robot talked to them. For experimental simplicity, the robot took initiative in conversation. It asked questions from a list of sentences prepared in advance, then it provided responses to what participants answered if it fitted with a fixed set of responding utterances. It continued such a question-answering dialog. This conversation routine was tele operated by an experimenter (conversation is out of the focus of this study, though this could be automated), while all the navigational framework was fully autonomous. The participants were asked to walk for 6 minutes with the robot.

6.3 Measurements

We recorded the trajectories of the robot and participant while walking together and computed the following objective measures for each participant to evaluate the system:

- *Relative angle* We computed the relative angle $\operatorname{angle} \left(\vec{p}_{t+1}^i p_t^i, \vec{p}_{t+1}^j p_t^j \right)$ from the robot to each participant for each time step, and took the average of all the time steps for each participant.
- *Relative distance* We computed relative distance $(|p_t^i - p_t^j|)$ for all time steps and averaged them.

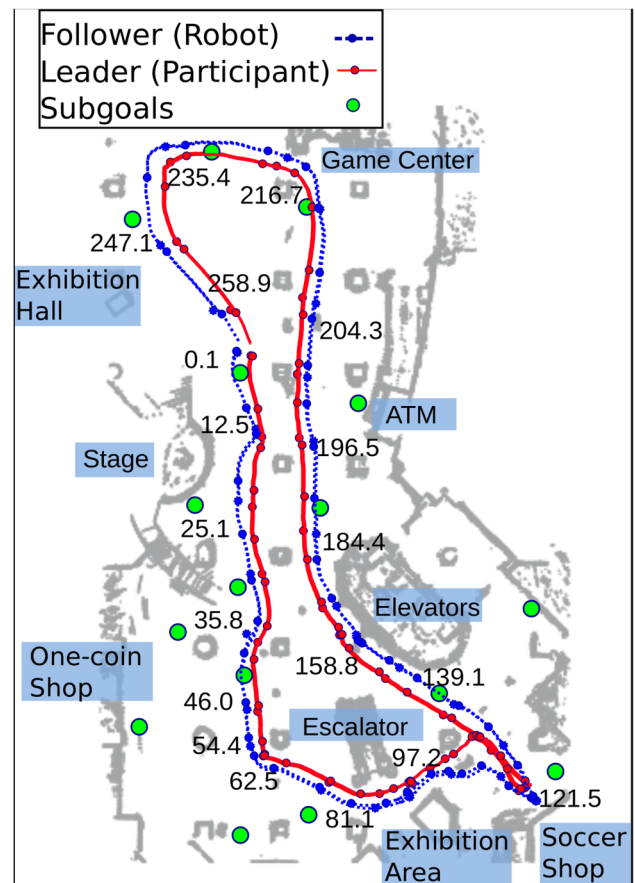


Fig. 14 An example of trajectories in a shopping arcade. The robot was controlled with the proposed method. The robot follower (robot) in dotted lines and leader in solid lines. Numbers denote time

For subjective evaluation, after each session, we asked participants to rate the following items on a 1–7 point Likert scale in a questionnaire:

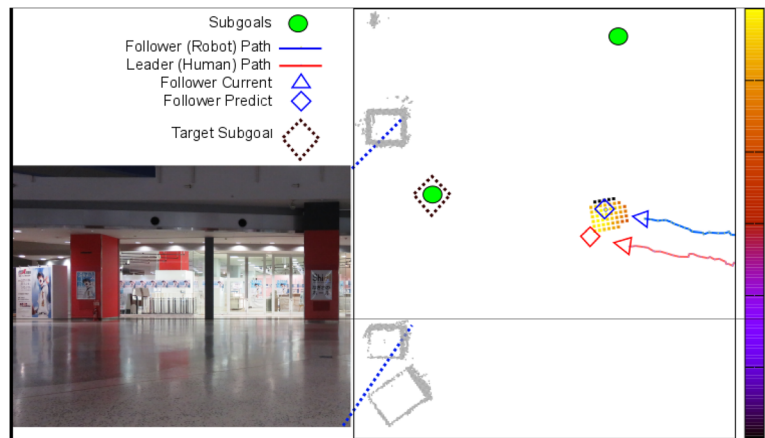
- Naturalness
- Perceived safety
- Overall evaluation

6.4 Results

1. Observations To introduce how the proposed system with subgoal selection worked, we present typical examples of its performance in a shopping arcade. Figure 14 shows an example of a person talking with the robot and walked side by side with the proposed method.

The person walked freely in the environment in a window-shopping context. He started walking through the main corridor in front of an exhibit stage (Fig. 14, $t = 54.5$), he passed in front of one-coin shop at $t = 46.0$ and then turned left passing around the escalators at $t = 81.1$. He then looked at an exhibition area at $t = 97.2$ arriving to a soccer goods

Fig. 15 Collaborative-state side-by-side walking toward a determined subgoal



shop where he turned back at $t = 121.5$ and kept walking. He then walked by in front of elevators looking at information signboards at $t = 158.8$, passing through ATM at $t = 196.5$, arriving to the game center at $t = 216.1$ where he turned left to arrive to an exhibition hall at $t = 247.1$.

In the figure, human and robot kept side-by-side most of the time as far as they moved straight, while at some turning points, e.g. at $t = 54.5, 97.2$, around $t = 158.8, 216.7$, the robot was slightly behind, as they were walking in leader–follower state and robot tried to open space. Once the robot was a bit away from person around $t = 97.2$, because the space was too narrow. Thanks to the knowledge about subgoal, the robot kept moving and meet again after once they split.

An example of computation in the collaborative-state is shown in Fig. 15. In the figure, the current trajectories of the human partner and the robot are shown in red and blue, the computation results for utilities (only for the robot) are shown in front of the robot’s current location with the grids with a range of colors from purple to yellow (yellow represents the highest in utility). At this moment, they are heading toward a subgoal located between the two red columns shown in the picture. The robot predicts that the person will move straight to this subgoal that is located between these two columns, and plans to move alongside the person. The robot successfully walks side-by-side with this person heading straight toward a subgoal.

An example of computation in the leader–follower state is shown in Fig. 16. While they walked straight toward one subgoal between two columns, there is another possible subgoal on their left. Since it has some transition probability, it remains within the set of possible subgoal. Thus, the robot is opening space for the leader in case he would turn towards the possible subgoal close to the ATM

In addition, we observed that velocity-based approach sometimes did not perform well. Figure 17 shows one of the typical scene including success and failure of the velocity-based approach. The robot was successful in mov-

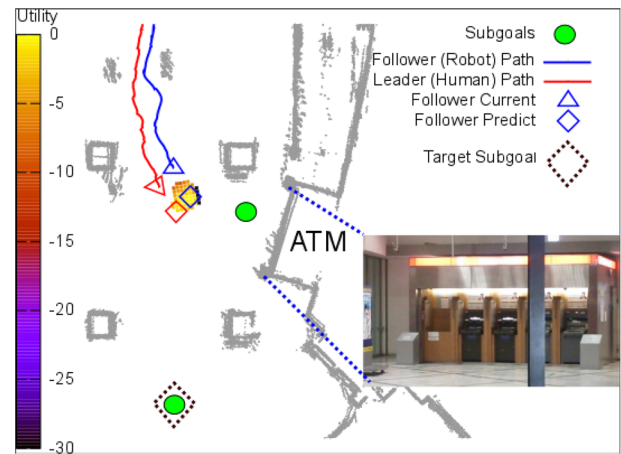


Fig. 16 Leader–follower state. The follower opens space as leader might turn to subgoal at the side

ing side-by-side formation in the left of the Fig. 17. Like this moment, as far as a partner human goes only straight, even if reactive, velocity-based approach would keep pace with the human partner, as the robot would react to his velocity.

However, in a real shopping arcade environment, people normally do window-shopping, which requires plenty of turns and stops, which made the velocity-based approach fail. As the robot cannot turn or change velocity instantly (and cannot anticipate human’s behavior), it tends to overshoot when the person stops or changes directions. One of such moments is illustrated in the center of the Fig. 17, when they reached to the location having two columns, the human suddenly turns left. This turning is not that sudden if one knows the environment, because without turning, he would collide into the column. However, the robot failed to keep the formation. The robot overshoots and crosses over the human, failing to stay at side-by-side formation. If the robot was able to perform opening-space behavior, it could turn left keeping alongside with the person; but, the lack of the opening-space behavior resulted in the person’s sudden

Fig. 17 Walking with velocity-based approach. As the robot has no open-space behavior it heads towards the human as he suddenly turns left

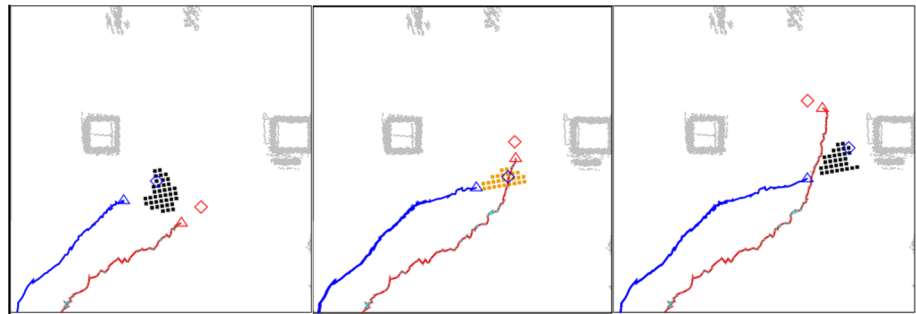
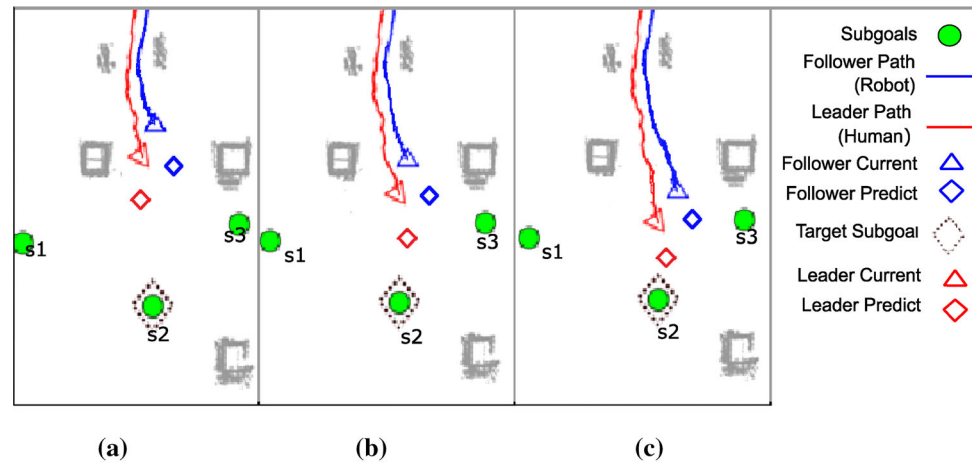


Fig. 18 Walking without subgoal selection. The robot conducts open space behavior all the time when multiple subgoals are available



turning behavior, and caused crossing over. We frequently observed such a crossing over in velocity-based approach. As a result, velocity-based approach tended to yield large relative angle and relative distance.

Figure 18 illustrates one of typical scene where the robot moved with “without the subgoal selection” method. As visible in the figure, the robot tended to stay behind, yielding smaller relative angle. When the robot was operating without the subgoal selection, we frequently observed the robot moving behind the human partner.

The reason for this behavior is due to the lack of subgoal selection mechanism. In the examples shown in the Fig. 18a–c, the system selected the subgoals s1, s2 and s3 as the possible subgoals, because their angles towards the moving direction were within the threshold. Transitions to subgoals in such locations could happen in other locations like a corner or junction in a corridor; however, in this location, transitions to s1 and s3 was less likely, which were observed as a small transition probability in pedestrian trajectories. Thus, in the proposed method (with subgoal-selection mechanism), s1 and s3 would be not chosen as the possible subgoals. In the “without the subgoal selection” method, since transition probability and transition radius were not used to filter out less likely subgoals, all subgoals were retained.

This resulted in the robot over-using opening-space behavior. When there are more than one possible subgoals, the system anticipates that there can be a change of direction.

Therefore, system goes into leader follower mode prompting the robot to make the adjustment anticipating a turn to a possible subgoal, in this case the subgoal s3. That is, the robot was anticipating that the person might turn towards s3, and stayed behind and slowed down. This would be the correct behavior in another place where it is more likely to anticipate people to turn in that direction.

6.5 Verification of Predictions 1 and 3

Figures 19 and 20 show the results for objective measures. A repeated-measures ANOVA was conducted that had within-participant’s variables of the experiment condition. Regarding *relative distance*, a significant main effect was revealed ($F(2,28)=117.584, p < 0.001, \eta^2 = 0.894$). Multiple comparisons among conditions were conducted with the Bonferroni method, which revealed that *relative distance* is shorter in the proposed model than the velocity-based condition ($p = 0.001$) and without the subgoal-selection condition ($p < 0.001$). No significant differences were found between the velocity-based condition and the without subgoal-selection condition ($p = 1.0$). Although the result of the *relative distance* in the proposed condition (1.25 m) is yet larger than that from the human pair average (0.815 m), it is closer to this one than other conditions.

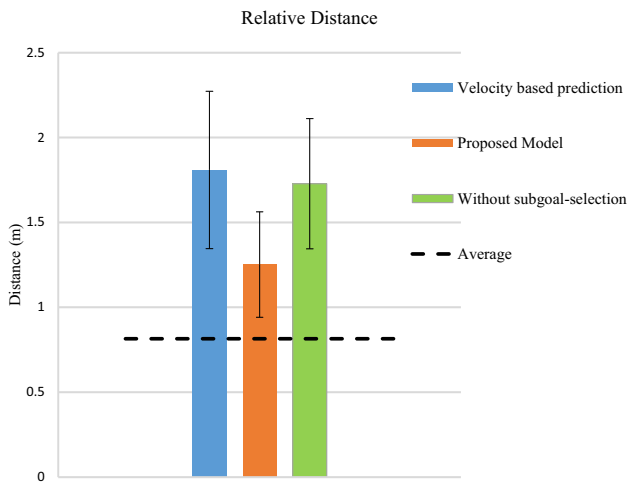


Fig. 19 Relative distance average obtained under each evaluation condition

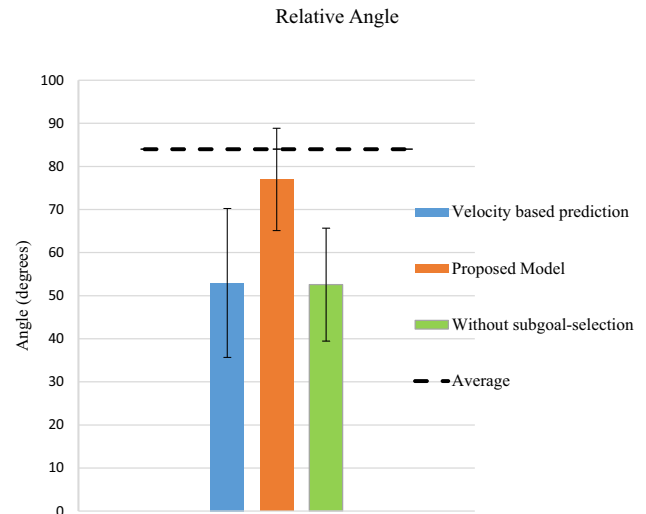


Fig. 20 Relative angle average for each evaluation condition

Regarding *relative angle*, a significant main effect was revealed ($F(2,28)=18.452, p < 0.001, \eta^2 = 0.569$). Multiple comparisons among conditions were conducted with the Bonferroni method, which revealed that *relative angle* is larger in the proposed model than the velocity-based condition ($p < 0.001$) and without subgoal-selection condition ($p < 0.001$). No significant differences were found between velocity-based and the without subgoal-selection condition ($p = 1.0$).

Although the result of the relative angle in the proposed condition (77°) is yet smaller than the one from human pair average (84°), it is closer to this one than other conditions. Therefore, as shown in Fig. 20 the proposed method performs best outperforming other two conditions.

As predicted, the robot provided relative distance and relative angle values closer to the values obtained in human side-by-side walking than the other two methods. Thus, Predictions 1 and 3 were confirmed.

6.6 Verification of Predictions 2 and 4

Figure 21 shows the results for subjective measures. A repeated-measures ANOVA was conducted that had within-participant's variables of the experimental condition. A significant main effect was revealed in *naturalness* ($F(2,28)=12.904, p < 0.001, \eta^2 = 0.480$), and *overall evaluation* ($F(2,28)=16.682, p < 0.001, \eta^2 = 0.544$). Main effect in *perceived safety* approached significance ($F(2,28)=2.634, p = 0.090, \eta^2 = 0.158$). Multiple comparisons among conditions were conducted for factors with significant main effects with the Bonferroni method, which revealed that *Naturalness* is higher in the proposed model than in the velocity-based condition ($p = 0.005$) and the without-subgoal-selection condition ($p = 0.013$). No significant

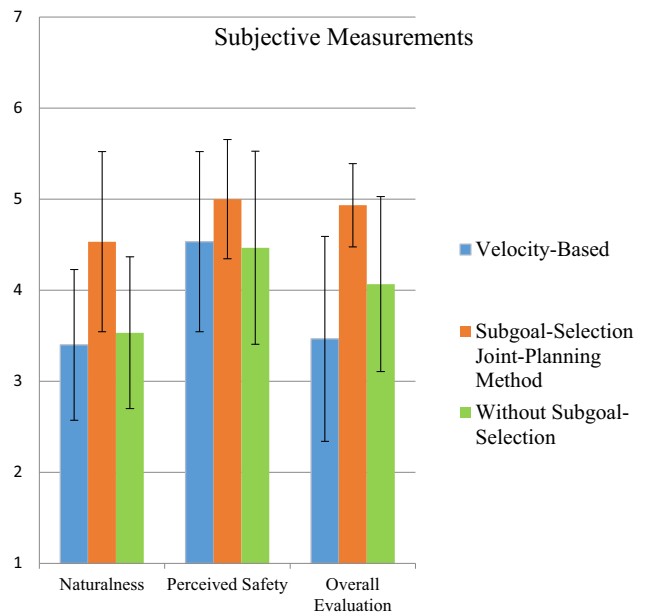


Fig. 21 Subjective evaluation results

differences were found between the velocity-based condition and the without-subgoal-selection condition ($p = 0.493$).

Overall evaluation is higher in the proposed model than the velocity-based condition ($p < 0.001$) and the without subgoal-selection condition ($p = 0.004$). No significant differences were found between the velocity-based condition and the without-subgoal-selection condition ($p = 0.135$).

Overall, predictions 2 and 4 were partly confirmed. As predicted, the robot provided naturalness and overall evaluation was better than the other two methods, though significant differences were not observed for perceived safety.

To make sure that the observed difference in verification of predictions 1 and 3 were because of the experimental condi-

tions and not because of different human behavior for taking different paths, we performed an analysis on the trajectories taken. Since participants moved freely in the environment, trajectories did not have the same length, therefore, we compare them based on subgoals visited and observing path ambiguity. We define as path ambiguity when at a point there are two or more possible subgoals achievable, thus allowing multiple possible paths.

In the velocity based system, method without subgoal selection and the proposed method participants visited 83.8, 80.2 and 82.1% of the available subgoals respectively. The computed path ambiguity for the human partner at the completion of each trajectory was 77.2, 72.4 and 74.1% respectively.

We conclude that participants walked in similar trajectories and results are comparable given that trajectories were similar in seen subgoals and in amount of ambiguity.

7 Discussion

7.1 Findings

There were differences between human–human data and human–robot motion results. Part of the reason regarding relative distance was due to our design. For safety reasons, during the planning, the system did not use the locations of anticipation grids nearby people, considering the time required to stop (only used the locations where it can move without causing collision). Thus, this is dependent with the robot hardware capability and safety design.

The experiment results did not show significance for safety impression. We consider that the example shown in Fig. 17 (from this study) illustrates an interesting contrast with the one shown in Fig. 3 (from [17]). In both moments, the person turned to the direction where the robot was moving. In Fig. 17, probably because there was enough space around, the person turned without any problems, but the robot instead crossed over. In Fig. 3 of [17], there is not so much space in the intersection, so the person needed to take quick turn hence nearly collide with the robot. Thus, we consider that such difference in environmental nature caused the difference result in perceived safety.

7.2 Generalizability

The proposed method was tested on a real shopping mall environment which contained a large open space with some defined corridors and columns within the environment (see Figs. 2, 5 and 14) which makes the modeling of the environment challenging as the topology of the environment is not evident. We think that our approach would be able to be gen-

eral enough to be used in other environments if the locations of subgoals are correctly placed.

Compared with other methods, due to the information related to subgoals, the robot could collaborate better with the walking person. However, the subgoal extraction method requires collecting trajectories from many people. Yet, note that for subgoal computation, we could have used other techniques for their extraction, or we could have just placed them by hand. As far as we have a set of locations towards which people tend to walk to, as well as probability values, we believe that our model would reproduce side by side walking.

Creating the accurate subgoals is of utmost importance for the generalization of the technique. However, in a much more dynamic environment the human motion pattern might change over a period of time. For example, a bargain sale or a discount offer may motive the people in the shopping mall to go in a different direction altering the normal motion pattern. Such dynamic aspect should be incorporated into the subgoal creation. One possible extension is that a higher weight be allocated to the recent trajectories while calculating the subgoals. Another mechanism is to integrate captured trajectories of each day into the subgoal calculation. This can be systematically done by capturing trajectories of people for a defined amount of time and using the captured trajectories in the subgoal calculation process.

In the case of having partial or slight changes in the environment, subgoal manual adjustment could be a simple and viable solution. If we know that the environmental differences are, manual adjustment could be simple and in some cases, it could be done based on common sense. For example, in simple corridor environments where people just go along the corridors and possibly turn at an intersection, thus, subgoals can be just placed in the intersection.

7.3 Limitations

Even if our robot has good speed and acceleration, its motion capabilities are not as quick and dynamic as walking humans. Nevertheless, although we identified some differences between human–robot motions, we reproduced the critical nature of side-by-side walking.

We had the case in which the robot mistakenly assigned an incorrect subgoal about 8% of the time. Nevertheless, such error was usually only momentarily, and overall the robot successfully coped with such errors.

The environment used in the experiment was a complex environment where there were limited number of other pedestrians at the time of the experiment. Considering the real use of the model, influence from other pedestrians is one major factor which should be considered and incorporated into the proposed model.

In addition, pedestrians when walking together often walk with different purposes and different environmental conditions. In the experiment conducted above we did not differentiate between the purposes. The participants as mentioned earlier were asked to walk freely rather than walking in any specific curve or straight line. Therefore, experimental results do not contain results which differentiate between environmental conditions and purpose. The participants were not necessarily young or elderly so there could be different results observed if the study was conducted based on age groups.

8 Conclusions

We developed a side-by-side walking model where the robot does not need to know the destination in advance. Based on human–human walking interaction data analysis, we developed a two-state model based on subgoals for side-by-side walking where one of the partners does not know the destination. The model was implemented with a mobile robot in a real shopping mall arcade. Subgoal locations and model parameters were calibrated using human walking data. Experimental results with human participants show that the proposed model was found more natural and was evaluated higher than the velocity–prediction approach and an approach without subgoal selection.

Acknowledgements This work was supported by JST, CREST. We thank Thomas Kaczmarek, Kanako Tomita and Chao Shi for their help in the realization of field experiments.

Funding This study was funded by JST, CREST.

Compliance with ethical standards

Conflicts of interest The authors declare that they have no conflict of interest.

References

1. Brscic D, Kanda T, Ikeda T, Miyashita T (2013) Person tracking in large public spaces using 3D range sensors. *IEEE Trans Hum Mach Syst* 43(6):522–534
2. Costa M (2010) Interpersonal distances in group walking. *J Non-verbal Behav* 34:15–26
3. Doucet A, Freitas N, Gordan N (2001) *Sequential Monte Carlo methods in practice*. Springer, Berlin
4. Garrell A, Sanfeliu A (2010) Model validation: Robot behavior in people guidance mission using DTM model and estimation of human motion behavior. In: *Proceedings of IEEE/RSJ international conference on intelligent robots and systems, Taipei, 2010*, pp 5836–5841. <https://doi.org/10.1109/IROS.2010.5651685>
5. Gockley R, Forlizzi J, Simmons R (2007) Natural person-following behavior for social robots. In: *Proceedings of the ACM/IEEE international conference on human–robot interaction (HRI)*. <https://doi.org/10.1145/1228716.1228720>
6. Helbing D, Farkas IJ, Molnár P, Vicsek T (2002) Simulation of pedestrian crowds in normal and evacuation situations. In: Schreckenberg M, Sharma MSSD, Sharma SD (eds) *Pedestrian and evacuation dynamics*. Springer, Berlin, pp 21–58
7. Helbing D, Molnár P (1995) Social force model for pedestrian dynamics. *Phys Rev E* 51(5):4282–4286. <https://doi.org/10.1103/PhysRevE.51.4282>
8. Hornung A, Wurm KM, Bennewitz M, Stachniss C, Burgard W (2013) OctoMap: an efficient probabilistic 3D mapping framework based on octrees. *Auton Robots*. <https://doi.org/10.1007/s10514-012-9321-0>
9. Hüttenrauch H, Severinson Eklundh K, Green A, Topp E (2006) A Investigating spatial relationships in human–robot interactions. In: *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*
10. Ikeda T, Chigodo Y, Rea D, Zanlungo F, Shiomi M, Kanda T (2012) Modeling and prediction of pedestrian behavior based on the subgoal concept. *Robot Sci Syst* 1:10
11. Kendon A (1990) Spatial organization in social encounters: the F-formation system. In: Kendon A (ed) *Conducting interaction: patterns of behavior in focused encounters*. Cambridge University Press, Cambridge, pp 209–238
12. Kobayashi Y, Kinpara Y, Takano E, Kuno Y, Yamazaki K, Yamazaki A (2011) Robotic wheelchair moving with caregiver collaboratively. *Advanced intelligent computing theories and applications*. *Asp Artif Intell* 6839:523–532. https://doi.org/10.1007/978-3-642-25944-9_68
13. Kuzuoka H, Suzuki Y, Yamashita J, Yamazaki K (2010) Reconfiguring spatial formation arrangement by robot body orientation. In: *Proceedings of the ACM/IEEE international conference on human–robot interaction (HRI)*. <https://doi.org/10.1145/1734454.1734557>
14. Moussaïd M, Perozo N, Garnier S, Helbing D, Theraulaz G (2010) The walking behaviour of pedestrian social groups and its impact on crowd dynamics. *PLoS ONE*. <https://doi.org/10.1371/journal.pone.0010047>
15. Morales Y, Satake S, Huq R, Glas D, Kanda T, Hagita N (2012) How do people walk side-by-side? using a computational model of human behavior for a social robot. In: *Proceedings of the ACM/IEEE international conference on human–robot interaction (HRI)*. <https://doi.org/10.1145/2157689.2157799>
16. Morales Y, Kanda T, Hagita N (2014) Walking together: side by side walking model for an interacting robot. *J Hum Robot Interact* 3(2):50–73. <https://doi.org/10.5898/JHRI.3.2.Morales>
17. Murakami R, Morales Y, Satake S, Kanda T, Ishiguro H (2014) Destination unknown: walking side-by-side without knowing the goal. In: *Proceedings of the 2014 ACM/IEEE international conference on human–robot interaction*
18. Nüchter A, Lingemann K, Hertzberg J, Surmann H (2007) 6d slam-3d mapping outdoor environments. *J Field Robot* 24(8–9):699–722. <https://doi.org/10.1002/rob.20209>
19. Prassler E, Bank D, Kluge B (2002) Key technologies in robot assistants: motion coordination between a human and a mobile robot. *Trans Control Autom Syst Eng* 4(1):56–61
20. Satake S, Kanda T, Glas DF, Imai M, Ishiguro H, Hagita N (2013) A robot that approaches pedestrians. *IEEE Trans Robot* 29(2):508–524. <https://doi.org/10.1109/TRO.2012.2226387>
21. Sisbot EA, Marin-Urias LF, Alami R, Simeon T (2007) A human aware mobile robot motion planner. *IEEE Trans Robot* 23(5):874–883
22. Thrun S, Burgard W, Fox D (2005) *Probabilistic robotics (intelligent robotics and autonomous agents)*. The MIT Press, London, pp 169–172
23. Walters ML, Dautenhahn K, Boekhorst R te, Koay K L, Kaouri C, Woods S, Nehaniv C, Lee D, Werry I (2005) The influence of sub-

- jects' personality traits on personal spatial zones in a human–robot interaction experiment. In: Proceedings of the IEEE international workshop on robot and human interactive communication (ROMAN). <https://doi.org/10.1109/ROMAN.2005.1513803>
24. Xu S, Duh HB-L (2010) A simulation of bonding effects and their impacts on pedestrian dynamics. *IEEE Trans Intell Transport Syst* 11(1):153–161. <https://doi.org/10.1109/TITS.2009.2036152>
 25. Zanlungo F, Kanda T (2013) Do walking pedestrians stability interact inside a large group? analysis of group and sub-group spatial structure. In: Proceedings of the 35th annual meeting of the cognitive science society (COGSCI), Berlin
 26. Zanlungo F, Ikeda T, Kanda T (2014) Potential for the dynamics of pedestrians in a socially interacting group. *Phys Rev E* 89(1):012811

Deneth Karunarathne completed a M.E. Degree specializing in Robotics in 2013 from the Osaka University's Intelligent Robotics Laboratory, Japan, and has since been enrolled as a Ph.D. candidate. Since 2012, he has been a student researcher at ATR's Intelligent Robotics and Communication Laboratory in Kyoto, Japan. His current research interests include autonomous navigation, sensor networks, and field robotics.

Yoichi Morales got M.E. and Ph.D. degrees in 2006 and 2009 from the University of Tsukuba's Intelligent Robot Laboratory in Tsukuba, Japan, where he also stayed half year as post-doctoral researcher until 2009. From 2009 to 2016 he was a researcher at the ATR Intelligent Robotics and Communication Laboratories in Kyoto, Japan where he is currently collaborative researcher. Since 2016 he is designated associate professor at the University of Nagoya. His current research interests include autonomous navigation, spatial cognition, perception and environment modeling. He is also a member of the IEEE, IEEE Robotics and Automation Society and the Robotics Society of Japan.

Takayuki Kanda received the B.E., M.E., and Ph.D. degrees in computer science from Kyoto University, Kyoto, Japan, in 1998, 2000, and 2003, respectively. He is currently a Group Leader with the ATR Intelligent Robotics and Communication Laboratories. His research interests include intelligent robotics and, human–robot interaction.

Hiroshi Ishiguro received a D. Eng. in systems engineering from the Osaka University, Japan in 1991. He is currently Professor of Department of Systems Innovation in the Graduate School of Engineering Science at Osaka University (2009-) and Distinguished Professor of Osaka University (2017-). He is also visiting Director (2014-) (group leader: 2002–2013) of Hiroshi Ishiguro Laboratories at the Advanced Telecommunications Research Institute and an ATR fellow. His research interests include sensor networks, interactive robotics, and android science. He received the Osaka Cultural Award in 2011. In 2015, he received the Prize for Science and Technology (Research Category) by the Minister of Education, Culture, Sports, Science and Technology (MEXT).