Robust Human Following by Deep Bayesian Trajectory Prediction for Home Service Robots

Beom-Jin Lee¹, Jinyoung Choi², Christina Baek³ and Byoung-Tak Zhang¹,²,³,⁴

Abstract—The capability of following a person is crucial in service-oriented robots for human assistance and cooperation. Though a vast variety of following systems exist, they lack robustness against dynamic changes of the environment and relocating to continue following a lost target. Here we present a robust human following system that has the extendability to commercial service robot platforms having a RGB-D camera. The proposed framework integrates deep learning methods for perception and variational Bayesian techniques for trajectory prediction. Deep learning modules enable robots to accompany a person by detecting the target, learning the target and following while avoiding collision within the dynamic home environment. The variational Bayesian techniques robustly predict the trajectory of the target by empowering the following ability of the robot when target is lost. We experimentally demonstrate the capability of the deep Bayesian trajectory prediction method on real-time usage, following abilities, collision avoidance and trajectory prediction of the system. The proposed system was deployed at the RoboCup@Home 2017 Social Standard Platform League and successfully demonstrated its robust functions and smooth person following capability resulting in winning the 1st place.

MULTIMEDIA MATERIAL
A video attachment to this work is available at: https://youtu.be/apRcOadMAo0.

I. INTRODUCTION
In the near future, humans collaborating with robots or robots assisting humans may become a common situation like smart-phones being deeply involved in our daily lives. However, for robots to be able to collaborate or provide assistance to humans, robust following of humans by a robot is very crucial. For example, transporting loads with human instruction, providing personalized service to customers, taking care of seniors or infants and even helping the family move their groceries. In this paper, we introduce a novel framework that achieves robust following of the humans for commercial domestic service robots. By integrating deep learning modules to perceive and learn robustly about the dynamic changing environment, using the Robot Operating System (ROS) to provide a universally adoptable action system and predicting the target’s trajectory to recover from failure of following, we believe that our framework could be adopted by most commercial home service robots which require robust following of the target person to provide personal services.

II. RELATED WORK
Human following by a robot has been an ongoing research topic in the robotics community [1], with annual robotic competitions [2], [3] to test the following performances. To achieve such an ability, previous studies worked with vision techniques to capture the human’s characteristic features to detect and track the human. For example, SIFT [4], ORB [5] and template matching [6] were used in human tracking. However, these approaches had several limitations with in illumination change, translation of objects and occlusion of the sensors. Moreover, the difficulty of separating a person between the foreground and the background was a very demanding issue to maintain a following system with a certain level of performance. Because of the difficulty in capturing features in the whole body image, methods which detect separate body parts of the human were studied. [7] used the face as the main clue for the robot to follow and [8] combined the features of the face and the legs to follow the person. However, these approaches suffered from the strict assumption that the person must be facing or heading towards the robot’s sensor for the robot to correctly follow the target. Consequently, other approaches which account the limitation of using only the directional (fixed) camera were proposed. They utilized other hardware that a robot could generally be equipped with (stereo camera, laser sensor, sonar, and thermal camera) to the human following systems. First, commonly used laser sensor based studies [9] were proposed and by [10], thermal camera was employed to detect the margin of the human to track them in an indoor environment such
as the hallway. However, with these methods accountable situation got smaller than the other studies, since each sensor had its own limitation on each situation. Therefore, combing such hardware were studied, as [11] used an omnidirectional camera and the lasers to figure out the environmental features while the laser’s point values were used to keep track of the person. Though combining new sensors has increased the following performance, the expense in using combined sensors has also increased. Moreover, not addressing the contextual information with the combinatorial information brought confusion in detecting and tracking the human.

As a result, to account the contextual information contained in the data, machine learning was employed into these systems. [12] used a predefined depth template matching technique where it could match the template with the input for the stereo depth camera, and built a support vector machine (SVM) based verifier to find the regions of the person to keep track of the person. However, in these studies, the experiment took place where there is no collision between the subject and the robot, ignoring the consideration of the situation where the robot could lose the target because the area was wide open and too simple for detection and following in a realistic environment. Moreover, for most studies, the operators were very robot friendly, as the operator walked really slow or shown intentional behavior for the robot to better track the person.

In contrast with the mentioned literature, in our framework, combining the high performance in recognition using deep learning methods, empowered by the computational power of GPU, and generally adoptable ROS system, we introduce a robust integrated system for home service robots to follow a person in the home environment. Our system contributes with 1) robust detection and identification of a person in real-time (around 0.3 s) in a homelike environment with state-of-the-art performance, 2) following the target with contextual information to perform better collision avoidance and 3) by recording a person’s coordinate trajectory in real-time matter, we could empower the robot with an ability to follow the person with variational Bayesian linear regression (VBLR) based trajectory prediction when the robot failed to continuously follow or lost the target person it was following.

Our Deep Bayesian Trajectory Prediction (DBTP) framework consists of three parts. First, the deep learning based perception module. Second, the controller selecting module where it switches the robot’s following behavior between the dynamics control, recursive path planning navigation control and reflex control for collision avoidance. Finally, the VBLR based trajectory prediction module.

In the following sections of the paper, we explain the overall structure of our proposed framework. Next, we present the experimental results designed to investigate the potential of our proposed framework in a real-life homelike environment. Also, we report the real-life usage of our proposed framework by participating in the RoboCup@Home Social Standard Platform League and winning the first place. Finally, in the concluding remarks, we discuss the possible directions for future investigations and improvements.

III. METHODOLOGY

To achieve robust following behavior of home service robots, we implemented a novel framework that involves the outlined steps: 1) real-time detection of people from the RGB-D image, where input is from the robot’s vision sensors, 2) identification and continuous learning of the following target, 3) estimation and prediction of the position and the trajectory of the target person for continuous following, and 4) selection of appropriate control (action) for robots to maintain the robust following behavior of the person.

A. Detecting and Learning to Follow a Person: Perception and Learning

1) Robot’s Real-time Perception System: To detect people in a real-time manner, we employed the YOLOv2 [13] algorithm. This algorithm has shown state-of-the-art performance of 78.6 mAP on VOC 2007 dataset, while still operating above real-time speed (3 ms) in object and person recognition. It is about 100x faster then any other existing object recognition algorithms such as the Faster RCNN [14].

2) Person Re-identification: The ability to identify the correct target is essential for home service robots to follow the target person and provide proper service to the correct person. Therefore, we investigated using a re-identification (one-to-one correspondence of database and the current image) algorithm with the detected bounding box of the person from the person detection module. We adopted the [15] re-identification algorithm which uses the Siamese network and combines the matching layers to achieve state-of-the-art performance. With this algorithm, we modified the algorithm to continuously learn the target person in an online manner. This improved the performance to 90% and also with the real-time process, errors such as the error affected by minimal noise could be ignored.
3) Target Trajectory Prediction (TTP): The target trajectory prediction (TTP) module is a novel method compared with conventional following algorithms. TTP considers the recovery mechanism when the robot loses the target person in the dynamic environment. Even in a simple homelike environment, many difficult areas or situations resulting in failure of the robot to follow could appear. For example, the environment structure formed in Fig. 6 shows when the target turns around the corner, they disappear. Therefore, this novel TTP module significantly improves the following ability in following within the dynamic environment. Our approach uses the variational Bayesian linear regression (VBLR) for predicting the future movement of the target with an online matter. The process of TTP is, first, from the bounding box of the identified target, it keeps recording the trajectory (coordinates) of the target every 0.2 seconds. With the history of trajectory, we use the following equations to learn and predict the target person’s trajectory.

VBLR is a high-dimensional sparse regression model [16]. The inputs are the coordinates of the trajectory \( x = x_{1:N} \) with the corresponding coordinate \( y = y_{1:N} \) and the weight vector \( w, \) where each components are D-dimensional.

The likelihood:

\[
p(y|x,w,\sigma) = \prod_{n=1}^{N} \text{Normal}(y_n|w^T x_n, \sigma)
\]

(1)

describes measurements corrupted by iid Gaussian noise with an unknown standard deviation \( \sigma \).

The prior on \( w \) and \( \sigma \) appears as the conjugate normal inverse-gamma

\[
p(w,\sigma|a) = \text{Normal}(w|0,(\sigma a)^{-1}) \Gamma a m a l(\sigma|a_0,b_0)
\]

(2)

where the prior \( \sigma \) appears as \( \sigma^{-1} \) in the variance of the zero-mean normal on \( w \).

Inference in variational Bayesian, calculate the posterior:

\[
p(w,\sigma,a) = p(w,\sigma|a)p(a) = \text{Normal}(w|0,\sigma(\text{diag}(\sqrt{a}))^{-1})
\]

\[
\Gamma a m a l(a|a_0,b_0) \prod_{i=1}^{D} \Gamma a m a l(a_i|c_0,d_0)
\]

(3)

the hyper-parameter \( a_0 = 1e-2, b_0 = 1e-4, c_0 = 1e-2, d_0 = 1e-4 \) is used for the calculation and the bound for learning is maximized by iterating over the updates for parameters \( w_N,a_N,b_N,c_N, \text{ and } d_N \) until the objective \( L(\cdot) \) reaches convergence. This enables the prediction of the lower-upper bound of the possible trajectory that the target would be appearing in. Additionally, the size of the trajectory data used were chosen by empirical experience from the experiments.

B. Robot Control for Following Person: Action

For the purpose of providing our framework as an open-source\(^1\), we connected all the modules with ROS for integrated control for the robots. To control the following person procedure, we implemented a pipeline with three different control flows. The dynamics control, recursive path planning navigation control and the reflex control.

1) Dynamics Control: From the identified result of the person, the robot receives the ROS message which contains the x-y coordinate of the bounding box of the person and also the closest distance of the person to the robot. From this, the robot controls its orientation (yaw) to robustly keep the target person at the center of the visual field and controls its velocity to maintain a constant distance to the target person being followed. This allows the robot to move forward when the target person moves forward, and move to a point near the person and stop when the person stops. Moreover, if the person approaches the robot too closely, it keeps a safe distance by backing off. We set a constant distance value of 1.2 m.

2) Navigation Control: Similar to the previously explained dynamics control of the environment, many previous studies on following depended on the dynamic control of the robot. This, however, resulted in making the target person walk in a very robot-friendly manner making it possible for the robot to successfully follow the target. The reason for this unnatural walking phenomena may be due to the unsteadiness of the perception system. The low consistency of the perceived information of the target made it difficult for the robots to follow when the target person was in a difficult path for the robot to follow. However, with the ROS message described in the previous section, the robot could also estimate the coordinate of the target in the map. When we say map, our framework does not require a predefined map but needs the coordinates system and localization of the robot. With the acquired coordinates of the target, we periodically plan the path for the robot to navigate in the dynamic environment. Here, if we use the map for localization or navigation, the location of the robot is confused with the sensed information. Therefore, we prefer not using the map. However, without the map and the periodical planning method, some situation where the robot gets stuck between the objects can occur which causes the running out of time to execute the periodically planned trajectory to escape from the situation. Therefore, we implemented a reflex module to avoid such situations and even avoid collision with the object, walls and other surrounding obstacles. This will be explained in the next section.

For the robot to properly plan a path near the target person and localize its position, we used the ROS Navigation Stack\(^2\) and the Adaptive Monte Carlo Localization (AMCL) method for SLAM. For planning, we used the default global planner and the Dynamic Window Approach (DWA) local planner for path planning to the designated position.

3) Reflex Control: Using only the dynamics control and the navigation control has limitations in collision avoidance and execution of during escaping in an isolated situation. Therefore, we implemented a reflex control where the robot senses a bump or a distance from the obstacle with laser or

\(^1\) https://github.com/soseazi/pal_pepper

\(^2\) http://wiki.ros.org/navigation
Fig. 3. a) Following the whole trajectory of the target person. b) Distance between robot and target person. The number indicates the step of following the target person.

any other sensors possible in the robot, for the robot to avoid the collision by backing up in the opposite direction of the obstacle and plan to move back in a curved, more smooth route to the designated position. This result will be discussed in the experiment section.

IV. EXPERIMENTAL RESULT

We have designed four experiments to demonstrate the proposed framework’s success in following the target person, avoiding collision and continuously following when the target person is lost, in a difficult situation in the environment (Fig. 6). Lastly, we report the results of our performance with this framework in the RoboCup@Home2017 following tasks. Videos will be provided by the supplementary materials and web-links.

A. INFRASTRUCTURE SETTING

We used two commercial robots to show the adaptability of our framework. One platform is the Softbank Pepper robot and the other is the Turtlebot2 with a height modified to 1 m for better capturing the environment (Fig. 2 top). Turtlebot2 was equipped with a Kinect like RGB-D camera Xtion and used bumper sensors to acquire data from the environment. The used laptop was the Asus EeePC 1215N laptop (Intel AtomTM D525 Dual Core Processor) to execute Turtlebot2. SoftBank Pepper’s specification can be found in the Pepper Wiki. The used sensors are the RGB camera, Xtion camera, laser sensors and bumpers.

For deep learning modules, we prepared a GPU server, Ubuntu 14.04 (ROS Indigo) based 12GB memory PASCAL.
Fig. 4. Infrastructure architecture; Tested mobile robots: SoftBank Pepper, Turtlebot2; Minimum required hardware specification are included.

GPU slotted computer. However around 6GB of memory was used to handle all the process.

For communicating the results between the server and the robots, we used 5 GHz Wi-Fi for consistent communication. The overall hardware architecture is depicted in Fig. 4. Also, it indicates the minimum requirement of each component.

B. Following

The first experiment is to test the overall performance of the home service robot following the target person. The experiment took place in a designed a homelike environment (Fig. 2 environment). As illustrated in Fig. 3, the robot’s trajectory (blue dot) is consistently following the person even when the person changes speed and direction. Moreover, at the dotted square X, Y, Z, the target person behaves with dynamic movements like wiggling side by side, moving in a narrow space and even moving toward the robot and going pass the robot. However, our system robustly follows the target person within 2.5m distance.

C. Collision Avoidance

To test whether our system could perform collision avoidance when following, we placed obstacles in the environment as depicted in Fig. 5. First, for the red box, the target person passes the obstacle very closely and quickly. In this case, the control system executed the dynamics control with the reflex module together to avoid the obstacle. The blue box obstacle in Fig. 5 was tested to see whether our action controller could avoid difficult situations of colliding with the obstacle. When the person went over the obstacle, it resulted in the obstacle being placed between the robot and the target person. For such a case, it is impossible for the robot to follow the target with only the dynamics control. However, our navigation control planned the path periodically in respect to the person’s distance and applied the reflex module when it approached close to the obstacle, resulting in the completion of following the person to the end.

D. Recovering Following When Lost Target

We examined our methods with two most difficult situations where the robot could easily lose the target person (Fig. 6). Task A is a situation when the person goes out the door and immediately turns right. This made our perception module capture the target person with a slight view in between the doorway (Fig. 7 solid lined box, top row [a,b,c]). Task B is when robot totally loses the perception of the target person, when the target person hides behind the wall by turning left (Fig. 7 dotted lined box, bottom row [d,e,f]). We compared our proposed VBLR with two other methods.

1) The Momentum Method: The momentum method calculates the target person’s coordinates by applying momentum to velocity and acceleration of the last few points, is shown in Fig. 7 a and d. The arc stripped line describes the wide variation where the target would be. This showed that this method had a very strong dependency on the last few
recognized coordinates of the target. Moreover, the predicted line (line with markers) could not even reach near the target point.

2) **Maximum Likelihood (ML) Method:** Maximum Likelihood is the well-known method in statistics where it selects the set of values of the model parameters that maximize the likelihood. Therefore, in our case, it provides the general trend of the data that is being represented by the trajectory history. On Fig. 7 b and e, the red strip-dotted line describes the predicted trajectory of the person. As shown in the figure, the robot could arrive outside. However, after arriving at the predicted point, it consumes some time to re-find the target person.

3) **Proposing Variational Bayesian Linear Regression (VBLR) Method:** For our VBLR method, the result greatly improved. Depicted in Fig. 7 c and f, the lower and upper bound of the trajectory predicts almost the exact coordinates of the target person. Possible improvements to enhance this may be a use of a method which learns the coordinate history size to be used to predict the target person’s trajectory.

4) **Time Consumption Comparison for Re-finding the Target Person:** The consumed time for finding the lost target person was measured. For the re-finding algorithm, we used the simple spinning around method until the robot finds the target. First, with task A, every method found the target. However, the gaps between each method were large in which our method achieved almost real-time re-following at that given situation. Moreover, for task B, the other two methods failed on detection of the target person. For the momentum method, the robot was unable to move out of the doorway. The ML method predicted the trajectory to go outside but went too far to recognize the target person. As a result, even for this task, our VBLR succeeded in going out of the doorway and finding the target within an average of 3 seconds. The average consumption time with 100 trials is illustrated in Fig. 8.

E. **RoboCup@Home Successful Following**

With our proposed system, we participated in the RoboCup@Home Social Standard Platform League as Team AUPAIR. This league is designed for the teams to test and compete for the many qualities including robot following the operator for home service robots with commercial robot platform (Pepper). In the league, there was a scenario called ‘help-me-carry.’ This scenario measured how much the robot could follow the target and provide services to the target operator. As presented in Fig. 9, our system adapted robot followed the operator very successfully when the operator walked very naturally. Moreover, we predicted the path to follow the target when the robot lost the operator while he passed through the doorway. Our robot kept following the predicted trajectory and succeeded in going out of the arena for continuous following. Also, we were the only team of 7 competing teams to point scores in this following task. The video of the results can be seen in the link written under the MULTIMEDIA MATERIAL section.
V. Conclusion

We proposed a robust following framework that could be adopted to commercial home service robots to follow the target person and provide personal services. The framework consisted of a perception-learning-action cycle to deal with the dynamic environment. However, a limitation of our work would be the dependency of using Wi-Fi and the range being limited to a home environment where the signal is available. However, as the communication technology is being improved every day, we predict that this dependency may not be a critical issue in the near future. We evaluated our performance with robust following, collision avoidance and prediction of target’s trajectory. The results were promising which proved the robustness of our framework. Also, by testing our framework on two different commercial robots, we have shown some adoptability of our framework. As the following methods improve, we anticipate a freely following home service robot to provide various personal services.

References