HI Robot: Human Intention-aware Robot Planning for Safe and Efficient Navigation in Crowds

Chonhyon Park[1,2], Jan Ondřej[1], Max Gilbert[1], Kyle Freeman[3], Carol O'Sullivan[1]

***Abstract*— We present an algorithmic framework for the early classification of human intentions, and use it to accurately predict future human motions when planning the path of a robot in an environment that is shared with humans. During an off-line learning phase, a classifier that can recognize when a human intends to interact with the robot is trained. At runtime, this trained classifier allows us to recognize humans who intend to interact with, or obstruct, the robot in some way. We validate our approach using both recorded and simulated data in an environment in which some humans intentionally obstruct the robot. Our classifier identifies these potential blockers, thus allowing the robot to safely and efficiently navigate the environment by minimizing the chances of being blocked.**

I. Introduction

In the past, robots have mainly been deployed in industrial environments in confined spaces where humans are typically not allowed. Recently, however, there is a trend towards the development of robots that perform tasks in the presence of humans, such as collaborative robots that share a person's workspace, or service robots that directly interact with their owners. Planning for the safe motion of such robots is therefore an important challenge and has consequently been an active research area [1], [2]. Many previous approaches assume a scenario in which the robot has to be aware of just the interaction subject or a few people in the workspace, as is typical for service robots.

More recently, however, robots have been appearing in public spaces such as streets or parks for service or entertainment purposes. In such environments, robots are required to be aware of multiple humans or even crowds in order to navigate safely. Several works have modelled this situation by considering the humans to be dynamic obstacles that the robot needs to avoid [3], [4], whereas others compute joint motions between the robot and the humans [5] based on the assumption that both the humans and the robot will try to avoid each other in a collaborative manner. This assumption is reasonable when considering service robots that assist in people's daily tasks. However, predicting different human motions using a single motion model may not be relevant in situations where the behaviors of the humans with respect to the robot are much more unpredictable.

Consider a robot that wanders around a public space during an event and interacts with guests for entertainment purposes. Some of the guests will almost certainly approach, as opposed to avoid, the robot and may even engage in potentially harmful activities. For example, it has recently been observed that children will sometimes exhibit abusive behavior towards a social robot and persistently obstruct its path [6]. Therefore, it is imperative that the robot can recognize such human intentions early in order to react appropriately and perform ameliorative actions.

Contribution: In this paper, we propose an algorithmic framework for the early classification of human intentions, and use it to accurately predict future human motions. Our approach uses an off-line learning phase based on the recorded trajectories of a robot and humans interacting in a communal space. Based on the features extracted from the human trajectories, we train a classifier to detect high-level human intentions to interact with the robot.

At runtime, we use the classification data provided from offline training with the state-of-the-art robot planner [5] to compute safe and efficient robot trajectories. We implement and analyze simulated scenarios as a proof-of-concept for our approach, where some humans intend to obstruct the robot, and the robot classifies humans likely to block and avoids them to minimize the chances of being blocked.

The rest of the paper is organized as follows: We give a brief overview of prior work on human-aware robot planning and navigation in crowds (Section II) and then present an overview of our planning algorithm (Section III). Off-line classifier training and runtime planning and prediction are described in Sections IV and V, and the performance of our algorithm in different scenarios is discussed in Sections VI and VII.

II. Related Work

In this section, we provide a brief overview of the most relevant prior work on human motion models for robot navigation and human intention recognition.

A. Human Motion Models for Robot Motion Planning

Predicting accurate future human motion is important for planning safe and efficient robot navigation and interaction with humans. Proxemic interpersonal distances [7] define the social norms of different human interactions and there has been extensive work on developing models to predict future human motions that are based on preserving these social distances. The problem is formulated using several different models, including potential and force computation of particles [8], velocity steering [9], and collision-free velocity computation [10], [11] of multiple humans. However, in all of these approaches the same motion model is applied to all agents, which is different from most real

[1] Disney Research Los Angeles, {chonhyon.park, jan.ondrej, max.gilbert, carol.osullivan}@disneyresearch.com

[2] University of North Carolina at Chapel Hill, chpark@cs.unc.edu

[3] Walt Disney Imagineering, R&D, Kyle.G.Freeman@disney.com

world scenarios in which a person's motion and reactions to the environment vary depending on individual characteristics. These interpersonal differences can lead to large variations in the motions of humans in a scene, as demonstrated by Guy et al. [12] for simulated agents with predefined heterogeneous traits.

Although a study of it has been shown that the same proxemic distances apply between humans and robots with a friendly demeanor [13], navigation algorithms for a robot in a crowd need to focus in particular on avoiding collisions between the robot and humans. The uncertainties which can be caused from the human motion model or imperfect sensing must therefore be taken into account. Thus, the future motion of a human is represented as a *Belief* state, which corresponds to the probability distribution over all possible motions. However, most robot navigation algorithms use the same motion model for all humans. Some approaches treat humans as stationary obstacles or assume that they are always cooperative with robots [14], [5]. On the other hand, Li et al. [15] proposes a robot motion planner which computes the safest robot trajectory by assuming the worst case scenario that all humans who react to the robot will hamper its motion.

Recently, online learning techniques with tracking filters have been integrated with motion models [16], [3], [5] in order to incorporate the past prediction error of individual humans into the prediction of future motions. Our work differs from these approaches in that we not only rely on a priori estimation, but also integrate it with the trained high-level intention information, thus facilitating the early classification of human intentions.

B. Human Intention Recognition

Human motion is usually caused by an individual's desire to achieve a particular goal or to perform an intended action, so the recognition of human actions or intentions has been a very active area of research [17], [18], [19]. Hansen et al. [20] use case based reasoning on human poses to estimate whether a human wants to interact with a robot, and demonstrate that the robot can be trained to react to different human behaviors with example-based learning. Chung and Huang [21] predict human behaviors based on their spatial properties (e.g., positions, orientations) relative to each other. Brscic et al. [6] (also mentioned in Sec. I) take a similar approach; by assuming that only unaccompanied children will perform abusive behaviors towards the robot, they compute the probability of abuse using the following features: (i) interaction time with robot; (ii) number of children present.

Other recent approaches use offline learning of intentions with additional information such as temporal relations between actions [22], [23] or object affordance [24], which results in early recognition of human actions. We use a learning-based method to detect the spatial relationships between the robot and humans, in order to react appropriately to the different human behaviors. For example, in our navigation approach, we use offline learning to detect when humans intend to block the robot, and use this information to compute a trajectory that avoids those humans who are likely to obstruct the robot's path.

Fig. 1: A snapshot[1] of our capture environment showing the robot being blocked intentionally.

III. Planning Framework Overview

In this section, we introduce the notation and terminology used in the rest of the paper and give an overview of our planning algorithm.

A. Notations and Assumptions

We assume that the robot (or the environment) has sensors with a known accuracy, which can track the position of multiple humans in the environment and use that information for realtime planning. The goal of our planning algorithm is to compute an efficient and safe robot trajectory to a given goal position while multiple humans are present in the robot's workspace. We assume that some of the humans can freely approach and interact with the robot. Based on this assumption, we detect humans who have the intention to interact with the robot based on their past trajectories. In our test-case, where the main interaction is to block the robot, the robot's path is re-planned by avoiding such humans in order to minimize the chances of being blocked.

We denote a robot's position as $\mathbf{q}_0$ and its trajectory as $\mathbf{Q}_0(t)$, which is a function of time t. The robot trajectory is also represented as a matrix $\mathbf{Q}_0$, which consists of discretized robot positions $\mathbf{q}_0^t$. We denote human trajectories as $\mathbf{Q}_1(t)$, ..., $\mathbf{Q}_N(t)$, or $\mathbf{Q}_1$, ..., $\mathbf{Q}_N$, where N is the number of humans in the robot workspace. We denote the trajectory set $\{\mathbf{Q}_0$, $\mathbf{Q}_1$, ..., $\mathbf{Q}_N\}$ as $\mathbf{Q}_{ALL}$.

B. Gaussian Process

Classification and regression are machine learning problems which involve computing the input-output mapping function $\mathbf{Y} = \mathbf{f}(\mathbf{X})$ from an empirical data set, where $\mathbf{X}$ denotes a set of input features, i.e., $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_M]^T$, and $\mathbf{Y}$ is the corresponding output, $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_M]^T$. We use Gaussian process models [25] for (i) the classification of humans who block the robot path and (ii) the regression of future human and robot trajectories based on their past trajectories.

[1] Intentionally blurred to prevent subjects' recognition.

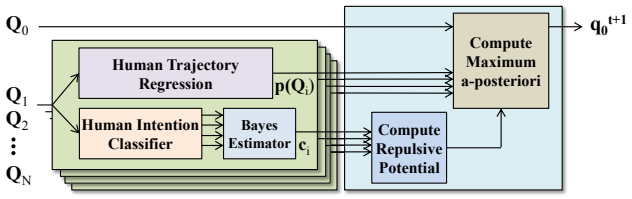


Fig. 2: An overview of the runtime planning framework. At each time step, the tracked past human trajectories are used to compute the probabilistic distributions of the trajectories and to classify the human blocking intentions. We compute the joint probabilistic distribution of all trajectories with the repulsive potential computed from the estimated human blocking intention. The robot position at the next time step $\mathbf{q}_0^{t+1}$ is computed from the robot trajectory that maximizes the joint trajectory probability.

A Gaussian process is defined as a collection of random variables, which have a joint multivariate Gaussian distribution. A Gaussian process $\mathbf{f}(\mathbf{X})$ can be specified using a mean function $\mathbf{m}(\mathbf{X})$ and a covariance function $\mathbf{k}(\mathbf{x}, \mathbf{x}')$, as

$$\mathbf{f}(\mathbf{X}) \sim GP(\mathbf{m}(\mathbf{X}), \mathbf{k}(\mathbf{x}, \mathbf{x}')). \tag{1}$$

Unlike other learning techniques, the Gaussian process does not rely on a specific function model, but computes a function $\mathbf{f}(\mathbf{X})$ by maximizing the smoothness of the covariance in terms of the covariance function $\mathbf{k}(\mathbf{x}, \mathbf{x}')$. Therefore, the selection of the covariance function $\mathbf{k}(\mathbf{x}, \mathbf{x}')$ is important for the Gaussian process, while the mean function is usually ignored and defined as a zero function without loss of generality. We will discuss the details of the covariance function selections and the computation of the outputs for classification and regression in Sections IV and V, respectively.

C. Human Intention-aware Planning

Our planning framework is shown in Fig. 2, which results in a robot trajectory that avoids humans who have intentions to approach the robot and block its path. We use a human intention classifier that is trained using supervised offline learning to compute the likelihood that the corresponding human will interfere with the robot motion. We compute the classification results for each past position $\mathbf{q}_i^t$ in a human trajectory $\mathbf{Q}_i$, then we compute the cumulative probability c_i from the individual classification results using a Bayes estimator.

In order to navigate in a populated environment, we replan the robot trajectory to avoid the blocking humans. We track the position trajectories of the humans $\mathbf{Q}_1$, $\mathbf{Q}_2$, ..., $\mathbf{Q}_N$ with the robot trajectory $\mathbf{Q}_0$. At each time step t, we compute the probabilistic distribution of each individual human trajectory $\mathbf{p}(\mathbf{Q}_i)$. We use Interaction GP (IGP) [5] as the underlying navigation algorithm to compute the robot trajectory. IGP computes the joint probability of the robot and human trajectories from the independently predicted trajectories $\mathbf{p}(\mathbf{Q}_i)$, by introducing the repulsive potential between trajectories. This joint probability is solely computed from the distance between trajectories, and assumes cooperative behaviors toward the robot for all humans. IGP does not consider the possible difference in the human intentions for the robot interactions. We formulate the repulsive potential $\phi()$ to be also affected by the classification results c_i, which lowers the joint distribution probability as c_i has a higher value. Therefore a high blocking likelihood c_i of the i-th human causes the robot to avoid the corresponding human with more space than others. The joint probabilistic distribution of all trajectories, $\mathbf{p}_{HI}(\mathbf{Q}_{ALL})$ is computed as:

$$\mathbf{p}_{HI}(\mathbf{Q}_{ALL}) = \phi(\mathbf{Q}_{ALL}, c_1, c_2, ..., c_n) \prod_{i=0}^{N} \mathbf{p}(\mathbf{Q}_i). \tag{2}$$

From $\mathbf{p}_{HI}(\mathbf{Q}_{ALL})$, we compute the maximum a posteriori robot trajectory $\mathbf{Q}_0^*$, that maximizes the probability of $\mathbf{p}_{HI}(\mathbf{Q}_{ALL})$, and compute the robot position at the next time step $\mathbf{q}_0^{t+1}$ on $\mathbf{Q}_0^*$.

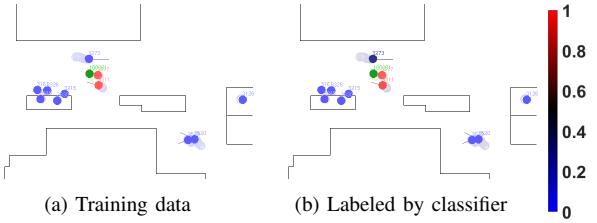
IV. Learning Human Intentions

In this section, we provide the details of the off-line training for human intention detection from recordings of real humans interacting with a robot.

A. Recording of Training Data

Our training data is collected from a communal kitchen area within a work environment (Fig. 1). Employees visit the kitchen to use the facilities and to converse with others, or simply walk through the area without stopping.

In order to collect the empirical data for the human intention classifier, we run experiments with a real robot. The robot we used in the experiments is a custom wheeled robot with a friendly demeanor. It has an integrated 1D LIDAR sensor attached to the front of the body near the ground to detect human feet, animals or other small objects. We programmed our robot to repeatedly follow a predefined path and to perform simple collision avoidance or stop its movement when a blocking obstacle is detected nearby. When stopped, the robot detours from its path to the next waypoint. As a safety measure, an operator with an emergency e-stop oversees the experiment.



(a) Training data (b) Labeled by classifier

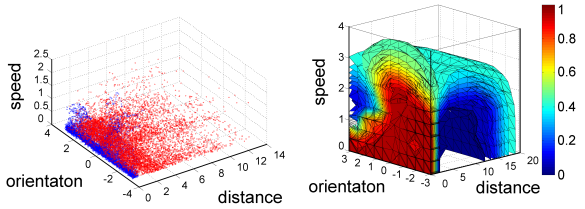
Fig. 3: (a) Trajectories of the robot (green) and humans who do (red) or do not (blue) block the robot. (b) Trajectories correctly labeled by our classifier.

In addition to the robot's local sensing, four Velodyne LIDAR sensors installed on the walls are used to detect the 3D positions of humans. An in-house algorithm is used

to detect the head and shoulder silhouette and the person's 3D position is calculated as the center of the head. The positions of the humans and the robot are collected in a sensing framework with a common coordinate system. These positions were recoded and used in our classifier for training and evaluation. For our supervised learning of human intentions to interact with the robot, we recorded two sessions with a total length of 2.5 hours from which we extracted over 3000 trajectories. The number of people present in the kitchen during the recording ranged between 1 and 20.

We annotated segments of each human trajectory where the robot was approached and consequently stopped, and set the human position samples in such segments as positive samples, and other positions as negative samples. For training, we used 7789 samples of positive (blocking) human positions and the same number of negative (non-blocking) samples randomly picked from the remaining trajectories. Some examples of annotated trajectories are shown in Fig. 3(a).

B. Classification of Human Intentions



(a) Input feature data (b) Human-intention classifier

Fig. 4: (a) The input data of the human intention classifier: relative positions, relative orientations, and absolute speeds. Blue and red dots represent positive and negative sets, respectively. (b) The plot of the trained classifier: 0.0 represents no blocking intentions, and higher values indicate an increased probability of blocking.

We assume that the intention of blocking the robot can be detected from the past trajectory of humans before the actual blocking happens, and use a Gaussian process to generate a classifier for such intentions. As the goal is to use the classifier to plan the robot's path, we transformed all the recorded trajectories into the robot-centric view. Among the multiple candidate input features that can be extracted from a position in the human trajectories, we choose our feature vector $\mathbf{x}$ to contain the *relative positions*, the *relative (velocity) orientations*, and the *absolute speeds* of past positions, which are invariant to environment coordinates, and also maximize the accuracy of the trained classifier (see Section VI). The training input data and trained classifier are plotted in Fig. 4.

As described in Section III-B, the Gaussian process is mainly computed using the covariance function $\mathbf{k}(\mathbf{x}, \mathbf{x}')$. There are commonly used covariance functions which are solely used or combined to capture the different characteristic of the input data (e.g., linear or non-linear) [25]. The training of the Gaussian process classifier aims to compute the optimal *hyperparameters* of those covariance functions which best fit the input data. For the classification of human blocking intentions, we choose the combination of the squared exponential, which is known as the de-facto default kernel for the classification due to its universal property that it can integrate different functions [26], and the noise kernel, which can deal with the position errors due to the sensor noise, as the covariance function:

$$\mathbf{k}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{(\mathbf{x}-\mathbf{x}')^2}{2l^2}\right) + \sigma_n^2 \delta(\mathbf{x}, \mathbf{x}'), \quad (3)$$

where l, σ_f^2 and σ_n^2 are hyperparameters which are optimized, and $\delta(\mathbf{x}, \mathbf{x}')$ is a Kronecker delta function which is 1 for $\mathbf{x} = \mathbf{x}'$ and 0 otherwise.

With M_c (7789 x 2) input features $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{M_c}]^T$, and their known output $\mathbf{y} = [y_1, y_2, ..., y_{M_c}]^T$ which are set to 1 for blocking human positions and 0 for non-blocking positions, the Gaussian process predicts the output y_* of $\mathbf{x}_*$ as

$$p(y_*|\mathbf{y}) \sim \mathcal{N}(\mathbf{K}_*\mathbf{K}^{-1}\mathbf{y}, \mathbf{K}_{**} - \mathbf{K}_*\mathbf{K}^{-1}\mathbf{K}_*^T), \quad (4)$$

where $\mathbf{K}$, $\mathbf{K}_*$, and $\mathbf{K}_{**}$ are defined as

$$\mathbf{K} = \begin{bmatrix} \mathbf{k}(\mathbf{x}_1, \mathbf{x}_1) & \mathbf{k}(\mathbf{x}_1, \mathbf{x}_2) & \cdots & \mathbf{k}(\mathbf{x}_1, \mathbf{x}_{M_c}) \\ \mathbf{k}(\mathbf{x}_2, \mathbf{x}_1) & \mathbf{k}(\mathbf{x}_2, \mathbf{x}_2) & \cdots & \mathbf{k}(\mathbf{x}_2, \mathbf{x}_{M_c}) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{k}(\mathbf{x}_{M_c}, \mathbf{x}_1) & \mathbf{k}(\mathbf{x}_{M_c}, \mathbf{x}_2) & \cdots & \mathbf{k}(\mathbf{x}_{M_c}, \mathbf{x}_{M_c}) \end{bmatrix},$$
$$\mathbf{K}_* = \begin{bmatrix} \mathbf{k}(\mathbf{x}_*, \mathbf{x}_1) & \mathbf{k}(\mathbf{x}_*, \mathbf{x}_2) & \cdots & \mathbf{k}(\mathbf{x}_*, \mathbf{x}_{M_c}) \end{bmatrix},$$
$$\mathbf{K}_{**} = \mathbf{k}(\mathbf{x}_*, \mathbf{x}_*) \text{ (see [25])}. \quad (5)$$

C. Runtime Human Intention Classification

When training the human intention classifier, we compute the probability of blocking intentions from a single past position. From k_1 past positions in a human trajectory $\mathbf{Q}_i$, the classifier computes k_1 probability values [c^{t-k_1+1}, c^{t-k_1+2}, ..., c^t]. For computing the cumulative blocking probability c_i of the trajectory $\mathbf{Q}_i$ from [c^{t-k_1+1}, c^{t-k_1+2}, ..., c^t], we consider two criteria: (i) handling sudden appearances of humans; and (ii) giving higher weights for the latest time steps. In the former case, a human can suddenly appear in a position close to the robot for many reasons (e.g., sensing error or obstacle occlusion). When the past trajectory does not have enough data, the errors from the sensing noise and the classification model affect the results more. In the latter case, we use the classification results of multiple positions to improve the prediction accuracy from the classification errors, but the humans can change their blocking intentions during the motion. Therefore, it is desired to assign higher weights to the latest classification results than to the old position results. With these criteria, the blocking probability c_i of a trajectory $\mathbf{Q}_i$ is computed as a weighted sum of k_1 probability values and the default estimate (0.5):

$$c_i = \frac{\sum_{j=0}^{k_1-1} a^j \cdot c_i^{t-j} + 0.5 * w_e}{\sum_{j=0}^{k_1-1} a^j + w_e}, \quad (6)$$

where a is a weight attenuation ratio ($a < 1$), and w_e is the weight given to the default estimate.

V. Human Motion Prediction and Robot Navigation

In this section, we present the details of the runtime robot planning algorithm, which combines trajectory regression with the classification of human intentions.

A. Trajectory Regression of Humans and Robot

As described in Sec. III-C and Fig. 2, our robot navigation algorithm combines individual trajectory regression with human-intention classification to predict the trajectories using a joint probabilistic distribution.

For each tracked human trajectory $\mathbf{Q}_i$, we use an online Gaussian process regression to compute the probabilistic distribution of the trajectory $\mathbf{p}(\mathbf{Q}_i)$. At a time step t, we use the positions of the last k_1 time steps as the input of the regression to compute the probabilistic distribution of the positions of k_1 past and k_2 future time steps. The regression is computed as

$$p(\mathbf{Q}_i) = p(\mathbf{Y}_*|\mathbf{Y}) \sim \mathcal{N}(\mathbf{K}_*\mathbf{K}^{-1}\mathbf{Y}, \mathbf{K}_{**} - \mathbf{K}_*\mathbf{K}^{-1}\mathbf{K}_*^T), \tag{7}$$

where the regression inputs are time $\mathbf{X} = [t-k_1+1, t-k_1+2, ..., t]^T$ and positions $\mathbf{Y} = [\mathbf{q}_i^{t-k_1+1}, \mathbf{q}_i^{t-k_1+2}, ..., \mathbf{q}_i^t]^T$. We also redefine $\mathbf{K}$, $\mathbf{K}_*$, and $\mathbf{K}_{**}$ to appropriate forms by defining $\mathbf{x}_*$ as a vector $[t+1, t+2, ..., t+k_2]$ to compute $\mathbf{Y}_* = [\mathbf{q}_i^{t+1}, \mathbf{q}_i^{t+2}, ..., \mathbf{q}_i^{t+k_2}]$.

For the training of the human intention classifier, we use the squared exponential kernel function as the covariance, because we have no prior knowledge of the model (Section IV-B). However, there are known models for the estimation of dynamic human motion behaviors. We use a linear covariance function with a non-linear Matérn covariance function to capture both linear and non-linear motion behaviors, along with the noise covariance function for the sensor noise. The covariance function is defined as

$$\begin{aligned}\mathbf{k}(\mathbf{x}, \mathbf{x}') = &\frac{\mathbf{x}\cdot\mathbf{x}'+1}{\sigma_v^2} \\ &+ \left(1 + \frac{\sqrt{3}(\mathbf{x}-\mathbf{x}')}{l}\right)\exp\left(-\frac{\sqrt{3}(\mathbf{x}-\mathbf{x}')}{l}\right) \\ &+ \sigma_n^2\delta(\mathbf{x}, \mathbf{x}').\end{aligned} \tag{8}$$

We optimize the hyperparameters σ_v^2, l, and σ_n^2, with the recorded trajectories used in the offline classifier training.

The regression of the robot trajectory is computed in the same manner as in the case of the human trajectory. The only difference is that the robot has a known goal position. We add this position $\mathbf{q}_0^{goal}$ and the approximated arrival time t^{goal} to the regression inputs $\mathbf{Y}$ and $\mathbf{X}$ to incorporate it in the prediction to guide the future robot trajectory to the goal.

B. Human Intention-aware Robot Planning

The final step of the runtime robot planning is to compute the joint probabilistic distribution of the trajectories $p_{HI}(\mathbf{Q}_{ALL})$, and extract the maximum a posteriori robot trajectory $\mathbf{Q}_0^*$. As described in Section III-C, we extend IGP [5] to use the computed blocking likelihoods of humans to vary the repulsive potential of each human, whereas IGP assumes that all humans will always exhibit cooperative behavior toward the robot and applies the same repulsive potential. This extension allows the robot to navigate more efficiently in situations where some humans are not cooperative, as shown in Section VII-B.

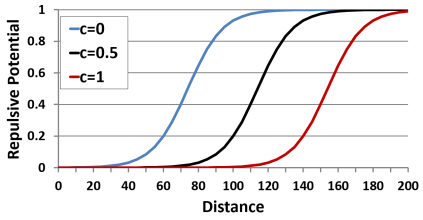


Fig. 5: The intention-aware repulsive potential curves $\phi(\mathbf{q}, \mathbf{q}', c)$ for $c = 0$, 0.5, and 1 with $w = 0.01$, $h = 10$, $s_0 = 120$ and $s_1 = 80$.

The repulsive potential $\phi(\mathbf{Q}_{ALL}, c_1, c_2, ..., c_N)$ in (2) is computed as

$$\phi(\mathbf{Q}_{ALL}, c_1, c_2, ..., c_n) = \prod_{i=1}^{N}\prod_{j=t+1}^{t+k_2}\phi(\mathbf{q}_0^j, \mathbf{q}_i^j, c_i) \tag{9}$$

$$\phi(\mathbf{q}, \mathbf{q}', c) = \frac{1}{1 + w\cdot e^{-\frac{(\mathbf{q}-\mathbf{q}')-s_0-s_1\cdot c}{h}}}, \tag{10}$$

where w and h are constants that determine the stiffness of the curve, and s_0 and s_1 determine the transition along the distance between the robot and humans. $\phi(\mathbf{q}, \mathbf{q}', c)$ is a sigmoid function translated along the distance between $\mathbf{q}$ and $\mathbf{q}'$. Based on Proxemic distances [7], we set s_0 and s_1 to reduce the probabilities of trajectories where the distance between the robot is less than the personal space ($d <= 120$ cm) when c is close to 1.0, while the robot is allowed to navigate the personal space (45 cm$< d <$ 120 cm) for small c values. Fig. 5 presents $\phi(\mathbf{q}, \mathbf{q}', c)$ for $c = 0$, 0.5, and 1.

Since the joint probability distribution $\mathbf{p}_{HI}(\mathbf{Q}_{ALL})$ computed from (2) is a non-Gaussian multi-modal distribution, we use stochastic approximation to compute the distribution. From the approximated $\mathbf{p}_{HI}(\mathbf{Q}_{ALL})$, we compute the joint trajectory $\mathbf{Q}_{ALL}^*$ that maximizes the probability, i.e.,

$$\mathbf{Q}_{ALL}^* = \arg\max_{\mathbf{Q}_{ALL}} \mathbf{p}_{HI}(\mathbf{Q}_{ALL}). \tag{11}$$

The robot position at the next time step $\mathbf{q}_0^{t+1}$ can be extracted from $\mathbf{Q}_0^*$ in $\mathbf{Q}_{ALL}^*$. This position maximizes $\mathbf{p}_{HI}(\mathbf{Q}_{ALL})$, which means it is the next position toward the goal while avoiding the humans by considering their blocking intentions. This position is used as the result of the planner and executed by the robot, and the planning is repeated at the next time step with the updated sensor information.

VI. Classification Results

The ability to accurately classify blockers is the novel and crucial part of our proposed motion planning framework. We labeled most of the blocking behaviors in the dataset and used them for training. To evaluate our classifier we used a 5-fold cross validation approach. We considered classification

	Predicted NO	Predicted YES
Actual NO	TN=7220	FP=569
Actual YES	FN=222	TP=7567

TABLE I: Confusion matrix of our human intention classifier. Classification is considered YES (blocking) for $c_i >= 0.5$ and NO for $c_i < 0.5$.

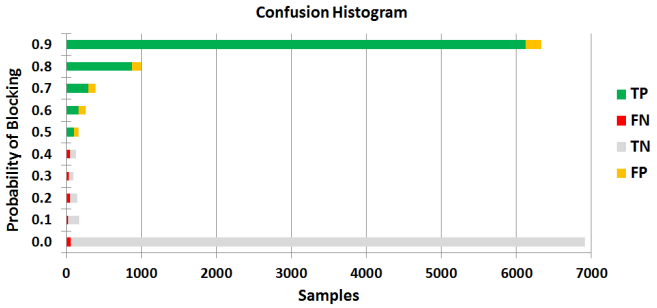
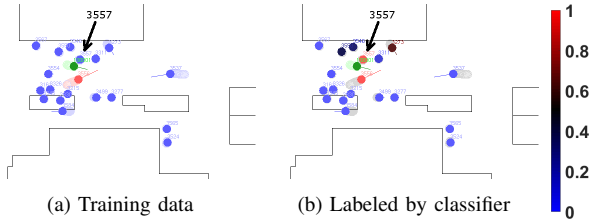


Fig. 6: A confusion histogram of the classifier for probability intervals with a 0.1 range. We plot the distribution of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) in different colors.



(a) Training data (b) Labeled by classifier

Fig. 7: False positive example in the intention classification: the person with id=3557 is not intentionally blocking the robot but is classified as a blocker with a high probability.

as positive (blocking) for probability $c_i >= 0.5$ and negative for $c_i < 0.5$. The classification c_i was computed from a single past position as described in Section IV-B. The overall achieved Accuracy is 0.95, Recall 0.97 and Precision 0.93. Table I shows the computed confusion matrix. The result of our classifier is the probability of blocking in the $[0, 1]$ interval and the confusion matrix captures only the overall binary accuracy. To better analyze the results we draw a confusion histogram of classifications for each 0.1 range, as shown in Figure 6.

The classifier has a much higher number of false positive (FP) samples than false negative (FN). This is mainly due to mis-classification of people who are passing by at a close distance, standing next to the robot or on the robot's path without intentionally blocking it. Figure 7 shows such an example. Even though these people are not blocking the robot intentionally, many of these cases can actually qualify as an obstruction of the robot's movement.

VII. Motion Planning Results

A. Methodology

We tested our motion planner's ability to circumvent blockers on the training dataset. This data was used during simulation to generate independent blocking events. We picked events where a human blocked the robot intentionally during the real-life data collection. First, the simulated robot followed the recorded path until 1/6s before the blocking event. Then, the planner was enabled with a goal point set on the recorded path 4s after the blocking event. Finally, when the simulated robot reached the goal, it was teleported back to the recorded path at that time stamp. This way, later human positions were not invalidated by a cumulative displacement of the robot. This method was similar to the technique used in previous work [27], where the robot replaced a pedestrian walking through a crowd in pre-recorded data.

Most metrics in robot-crowd motion planners fall into one of two categories: safety or efficiency. The number of times an emergency stop is called for the robot has been considered as a safety metric and the time taken to navigate through the crowd was used to measure efficiency. In previous work [27], the minimum distance the robot has been to a pedestrian has been used to measure safety and the length of the path for efficiency. Running our planner on a simulated robot with pre-recorded data, we chose a similar approach to the latter method, measuring the minimum distance to a pedestrian during a blocking event as well as the distance required to reach the goal during that blocking event. We recorded metrics for 34 blocking events across two sets of recorded data.

B. Evaluation

In order to validate the benefit of our planner based on human intention classification and to compare it with previous work that assumes unvarying human behaviors, we evaluate the performance of 4 robot planners which have different behaviors toward humans, in two benchmark scenarios. The recorded data in the first benchmark is of a much denser crowd with more blocking behavior exhibited, while the data for the second benchmark is more sparse, with only a few individual blockers occasionally trying to block the robot. We compare our human-intention aware planner with the following planners with static human intention prediction values, each of which emulate the different assumptions of previous works:

- $c = 0.2$: the planner assumes that every human is unlikely to block the robot.
- $c = 0.5$: the planner makes no assumptions about the humans.
- $c = 0.8$: the planner assumes that every human is likely to block.

The results shown in Table II demonstrate how our planner scales with increasing crowd density. In a less dense area, all the planners perform similarly because, regardless of the classification, they are at least weighted to avoid an individual blocker. In the dense benchmark however, our planner takes advantage of people who seem less likely to block and thus generates a more efficient path through the crowd. The robot avoids risky situations that may have led to a long delay, but at the same time it could exploit the knowledge of which pedestrians are unlikely to be blockers by cutting across in front of them based on the high probability that

Benchmark		Intention Classified	Not Classified		
			$c = 0.2$	$c = 0.5$	$c = 0.8$
Dense Crowd	Length	100.00%	110.53%	128.32%	163.99%
	Safety	0.5017	0.5454	0.5354	0.5192
Sparse Crowd	Length	100.00%	96.460%	97.530%	105.52%
	Safety	0.6321	0.5643	0.6222	0.6582

TABLE II: We evaluate the performance of our human intention-aware planner with 3 other planners having different assumptions on the human behavior in two benchmarks. The results demonstrate that in the dense situation, our planner computes the most efficient routes with only minimal reductions in the safety margin.

those pedestrians would not get in its way. This way, our planner achieves the most efficient routes around blockers on average, with only minimal reductions in its safety margin.

VIII. Conclusions and Future Work

In this paper we present an algorithmic framework that takes human intentions into account when planning safe and efficient trajectories for robots, in particular in crowded and unpredictable environments. We use an off-line learning process to train a classifier to detect human intentions to block the robot. At run-time, we use the trained classifier with regression based on past trajectories to predict where the human pedestrians will move to in the future. Our planning algorithm thus allows the robot to navigate amongst crowds of people more efficiently and safely. We validate our algorithm on different benchmark scenarios and compare with prior work using real recorded trajectories and a simulated framework.

There are many avenues for future work. So far, we have only used past human trajectories to predict their future motions. Additional features such as height [6], age, or gender can be used to improve the accuracy of the classification. Multiple class classification of humans rather than binary blocking and non-blocking can be used to model other human behaviors. Finally, we intend to integrate our planning algorithm into the real robot hardware architecture.

References

[1] N. E. Du Toit and J. W. Burdick, "Robot motion planning in dynamic, uncertain environments," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 101–115, 2012.

[2] J. Mainprice and D. Berenson, "Human-robot collaborative manipulation planning using early prediction of human motion," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2013, pp. 299–306.

[3] S. Kim, S. J. Guy, W. Liu, D. Wilkie, R. W. Lau, M. C. Lin, and D. Manocha, "Brvo: Predicting pedestrian trajectories using velocity-space reasoning," *The International Journal of Robotics Research*, 2014.

[4] S. Choi, E. Kim, and S. Oh, "Real-time navigation in crowded dynamic environments using gaussian process motion control," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 3221–3226.

[5] P. Trautman, J. Ma, R. M. Murray, and A. Krause, "Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 335–356, 2015.

[6] D. Brscić, H. Kidokoro, Y. Suehiro, and T. Kanda, "Escaping from children's abuse of social robots," in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 2015, pp. 59–66.

[7] E. T. Hall, *The hidden dimension*. Doubleday & Co, 1966.

[8] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.

[9] C. W. Reynolds, "Steering behaviors for autonomous characters," in *Game developers conference*, vol. 1999, 1999, pp. 763–782.

[10] C. Fulgenzi, A. Spalanzani, and C. Laugier, "Dynamic obstacle avoidance in uncertain environment combining pvos and occupancy grid," in *2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 1610–1616.

[11] J. van den Berg, S. Patil, J. Sewall, D. Manocha, and M. Lin, "Interactive navigation of multiple agents in crowded environments," in *Proceedings of the 2008 symposium on Interactive 3D graphics and games*. ACM, 2008, pp. 139–147.

[12] S. J. Guy, S. Kim, M. C. Lin, and D. Manocha, "Simulating heterogeneous crowd behaviors using personality trait theory," in *Proceedings of the 2011 ACM SIGGRAPH/Eurographics symposium on computer animation*. ACM, 2011, pp. 43–52.

[13] M. L. Walters, K. Dautenhahn, K. L. Koay, C. Kaouri, R. t. Boekhorst, C. Nehaniv, I. Werry, and D. Lee, "Close encounters: Spatial distances between people and a robot of mechanistic appearance," in *2005 5th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2005, pp. 450–455.

[14] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1726–1743, 2013.

[15] H. Li, O. Islas Ramirez, and M. Chetouani, "Potential human reaction aware mobile robot motion planner: Potential cost minimization framework," in *2014 RO-MAN: The 23rd IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2014, pp. 441–448.

[16] M. Kuderer, H. Kretzschmar, C. Sprunk, and W. Burgard, "Feature-based prediction of trajectories for socially compliant navigation." in *Robotics: science and systems*, 2012.

[17] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473–1488, 2008.

[18] W. Yang, Y. Wang, and G. Mori, "Recognizing human actions from still images with latent poses," in *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 2030–2037.

[19] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from rgbd images," in *2012 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 842–849.

[20] S. T. Hansen, M. Svenstrup, H. J. Andersen, and T. Bak, "Adaptive human aware navigation based on motion pattern analysis," in *The 18th IEEE International Symposium on Robot and Human Interactive Communication, 2009.* IEEE, 2009, pp. 927–932.

[21] S.-Y. Chung and H.-P. Huang, "A mobile robot that understands pedestrian spatial behaviors," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2010, pp. 5861–5866.

[22] S. Nikolaidis, P. Lasota, G. Rossano, C. Martinez, T. Fuhlbrigge, and J. Shah, "Human-robot collaboration in manufacturing: Quantitative evaluation of predictable, convergent joint action," in *44th International Symposium on Robotics (ISR), 2013.* IEEE, 2013, pp. 1–6.

[23] K. P. Hawkins, N. Vo, S. Bansal, and A. F. Bobick, "Probabilistic human action prediction and wait-sensitive planning for responsive human-robot collaboration," in *13th IEEE-RAS International Conference on Humanoid Robots (Humanoids), 2013.* IEEE, 2013, pp. 499–506.

[24] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE Transactions on Pattern Analysis and Machine Intelligence.*, vol. 38, no. 1, pp. 14–29, 2016.

[25] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. MIT Press, 2006.

[26] D. Duvenaud, "Automatic model construction with gaussian processes," Ph.D. dissertation, University of Cambridge, 2014.

[27] P. Trautman and A. Krause, "Unfreezing the robot: Navigation in dense, interacting crowds," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010.* IEEE, 2010, pp. 797–803.