Two People Walk Into a Bar: Dynamic Multi-Party Social Interaction with a Robot Agent

Mary Ellen Foster[1,*] Andre Gaschler[2] Manuel Giuliani[2]
Amy Isard[3] Maria Pateraki[4] Ronald P.A. Petrick[3]
[1] School of Mathematical and Computer Sciences, Heriot-Watt University, Edinburgh, UK
[2] fortiss GmbH, Munich, Germany [3] School of Informatics, University of Edinburgh, Edinburgh, UK
[4] Foundation for Research and Technology – Hellas (FORTH), Heraklion, Crete, Greece
M.E.Foster@hw.ac.uk {gaschler,giuliani}@fortiss.org
{amyi,rpetrick}@inf.ed.ac.uk pateraki@ics.forth.gr

ABSTRACT

We introduce a humanoid robot bartender that is capable of dealing with multiple customers in a dynamic, multi-party social setting. The robot system incorporates state-of-the-art components for computer vision, linguistic processing, state management, high-level reasoning, and robot control. In a user evaluation, 31 participants interacted with the bartender in a range of social situations. Most customers successfully obtained a drink from the bartender in all scenarios, and the factors that had the greatest impact on subjective satisfaction were task success and dialogue efficiency.

Categories and Subject Descriptors: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems – Evaluation/methodology; I.2.9 [Artificial intelligence]: Robotics – Operator interfaces

General Terms: Human Factors

Keywords: Social robotics, Multi-party interaction

1. INTRODUCTION

A robot interacting with humans in the real world must be able to deal with situations in which socially appropriate interaction is vital. It is not enough simply to achieve task-based goals: the robot must also be able to satisfy the social goals and obligations that arise during the course of human-robot interaction. In the JAMES project, we are addressing this issue by developing a robot bartender (Figure 1) which is able to deal with multiple customers in a dynamic setting. Interactions in the context of a bartending scenario incorporate a mixture of task-based aspects (e.g., ordering and paying for drinks) and social aspects (e.g., engaging in social conversation, managing multiple transactions), both of which present challenges: the robot bartender must be able to recognise, understand and respond appropriately to both the social and the task-based needs of the humans that it encounters, and to successfully distinguish between the two types.

*The author names are listed in alphabetical order.

Figure 1: The JAMES robot bartender

Interaction 1 (Socially inappropriate)	**Interaction 2 (Socially appropriate)**
One person, A, approaches the bar and turns towards the bartender	
Robot (to A): How can I help you?	Robot (to A): How can I help you?
A: A pint of cider, please.	A: A pint of cider, please.
A second person, B, approaches the bar and turns towards the bartender	
Robot (to B): How can I help you?	Robot (to B): One moment, please.
B: I'd like a pint of beer.	Robot: (Serves A)
Robot: (Serves B)	Robot (to B): Thanks for waiting. How can I help you?
Robot: (Serves A)	B: I'd like a pint of beer.
	Robot: (Serves B)

Figure 2: Social interaction in a bar setting

As a concrete example, consider the bartender interactions shown in Figure 2. At the end of both interactions, the needs of both customers A and B have been successfully met: each has made a request and has been served by the bartender. However, the second interaction is clearly more appropriate than the first. Not only are the customers served in the same sequence that they made their requests, but the robot also interacts with customer B in a more socially acceptable manner, by acknowledging B's arrival and completing the existing transaction before dealing with a new request. This demonstrates that, while many human-robot interactions may lead to the same goal at the task level, the quality of those interactions can be greatly enhanced by getting the "people skills" right.

Even a simple scenario like this one poses state-of-the-art challenges: the vision system must accurately track the locations and body postures of the two agents; the speech recogniser must be able to detect and deal with speech from multiple users in an open setting; the reasoning components must determine that each cus-

tomer requires attention and should choose appropriate behaviours to deal with both of them; while the output components must select and coordinate actions for all of the output channels that correctly realise the high-level plans, including both communicative actions and behaviours of the robot manipulators.

This work fits into the active research area of *social robotics*: "the study of robots that interact and communicate with themselves, with humans, and with their environment, within the social and cultural structure attached to their roles." [12]. Most current social robots play the role of a companion, often in a long-term, one-on-one relationship with the user [e.g., 7–9]. In this context, the primary goal for the robot is to build a relationship with the user through social interaction: the robot is primarily an interactive partner, and any task-based behaviour is secondary to this overall goal.

We address a style of interaction which is distinctive in two main ways. First, while most existing social robots deal primarily with one-on-one interactive situations, the robot bartender must deal with dynamic, multi-party scenarios: people constantly enter and leave the scene, so the robot must constantly choose appropriate social behaviour while interacting with a series of new partners. Second, while existing social robotics projects (even those that deal with multiple partners such as [24, 25]) generally take social interaction as the primary goal, the robot bartender supports social communication in the context of a cooperative, task-based interaction. Also, existing "robot bartenders" [e.g., 14, 23] focus on the physical tasks associated with bartending (i.e., actually preparing and serving drinks), and fail to consider the social context.

Our robot bartender is most similar to the multimodal interactive kiosk described by Bohus and Horvitz [6], which handles situated, open-world, multimodal dialogue in scenarios such as a reception desk. Their system incorporates models of multi-party engagement, turn-taking, and intention recognition, and has been evaluated in a series of real-world and laboratory studies. The robot bartender extends this work by adding physical embodiment, which has been shown to have a large effect on social interaction: for example, physical agents have been found to be more appealing, perceptive, and helpful than virtual agents [36], and to result in more positive and natural interactions [2, 22].

2. SYSTEM DETAILS

As shown in Figure 1, the bartender robot consists of two manipulator arms with humanoid hands mounted in a position to resemble human arms, along with an animatronic talking head. The software architecture (Figure 3) uses a standard three-layer structure: low-level components deal with modality-specific, highly detailed information such as spatial coordinates, speech-recognition hypotheses, and robot arm trajectories; the mid-level components deal with abstract, cross-modal representations of states and events; while the high-level components reason about the most abstract structures, such as knowledge and actions represented in a logical form.

On the input side, the low-level components include a vision system (Section 2.1), which tracks the real-time location of all people in the scene as well as their body language, along with a linguistic processing system (Section 2.2) combining a speech recogniser with a natural language parser to create symbolic representations of the spoken contributions of all users. The low level also includes output components (Section 2.5) that control the animatronic talking head (which produces synthesised speech, facial expressions, and gaze behaviour) and the robot arms and hands (which can point at and manipulate objects), along with a robot simulator.

The primary mid-level input component is the social state manager (Section 2.3), which combines information from the low-level input components to estimate the real-time social and communicative state of all users. On the output side, the output planner (Section 2.2) both translates the selected communicative and task-based acts into specific action sequences for the low-level components and also coordinates the execution of those sequences.

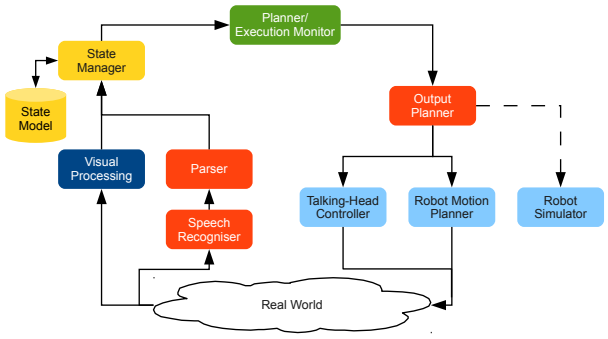

Figure 3: System architecture

Finally, the high level includes a knowledge-level planning component (Section 2.4) that generates plans for the robot to achieve its goals, where the plans include a mixture of domain actions (e.g., manipulating objects in the world), sensing actions (e.g., using the robot arms to test object properties), and communicative actions (e.g., getting a customer's drink request). The high-level system also includes a plan execution monitor which tracks the execution of the plan steps by monitoring the state of the world and updates the state of the plan and/or re-plans as necessary.

2.1 Visual processing

The vision module utilises input from visual sensors to detect and track in real time the faces and hands of people in the scene and to extract their 3D position, and also to derive each person's focus of attention via torso orientation.

To detect and track faces and hands we employ and extend a blob-tracking approach [3], according to which foreground, skin-coloured pixels are identified according to their colour and grouped together into skin-coloured blobs. Information about the location and shape of each tracked blob is maintained by means of a set of pixel hypotheses which are initially sampled from the observed blobs and are propagated from frame to frame according to linear object dynamics computed by a Kalman filter. The distribution of the propagated pixel hypotheses provides a representation for the uncertainty in both the position and the shape of the tracked object.

Moreover, an incremental classifier has been developed [4, 26] which extends the above blob tracking approach and which is used to maintain and continuously update a belief about whether a tracked hypothesis of a skin blob corresponds to a facial region, a left hand or a right hand. For this purpose, we use a simple yet robust feature set which conveys information about the shape of each tracked blob, its motion characteristics, and its relative location with respect to other blobs. The class of each track is determined by incrementally improving a belief state based on the previous belief state and the likelihood of the currently observed feature set. To derive the 3D position of the centroids of coloured regions, we apply the above tracking approach to precalibrated stereo images, establish correspondences of the detected coloured regions in the stereo images by using simple, computationally inexpensive techniques [1], and extract the 3D positions in a world-centred coordinate system. Figure 4 shows the output of the face and hand tracking process.

Also of interest in this domain is the focus of attention of a person approaching the bar, which is derived from information on torso orientation (arm tracking). For the torso orientation, a tracking approach is used [34] to track both arms (four parameters for

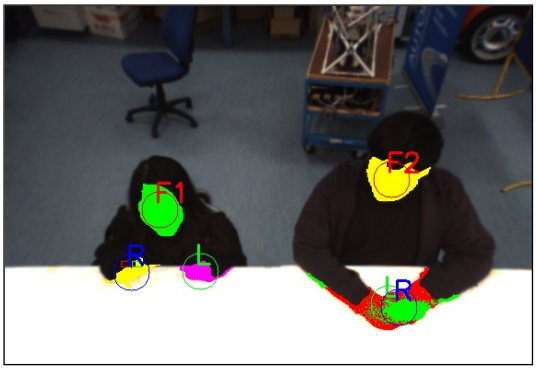

Figure 4: Output of face and hand tracking

```
<lf>
  <node id="w1" pred="can-verb" mood="int" tense="pres"
        voice="active">
    <rel name="Body">
      <node id="w3" pred="get-verb">
        <rel name="ArgOne">
          <node id="w2" pred="pron" num="sg" pers="1st"/>
        </rel>
        <rel name="ArgTwo">
          <node id="w5" pred="coke" det="indef" num="sg"/>
        </rel>
      </node>
    </rel>
    <rel name="HasProp">
      <node id="w0" pred="please"/>
    </rel>
  </node>
</lf>
```

Figure 5: OpenCCG logical form for "Please can I get a Coke?"

each arm) as well as the orientation of the human torso (one parameter). To reduce the complexity of the problem and to achieve real-time performance, the model space is split into three different partitions and tracking is performed separately in each of them. More specifically, a Hidden Markov Model tracks the orientation of the human torso in the 1D space of all possible orientations, and two different sets of particles are used to track the four degrees.

In later versions of the system, the vision module will detect and track objects used in different human actions—such as pointing, looking at, grabbing, and holding an object—by extending the blob-tracking approach to handle multiple colour-classes. An additional attentive cue, that of head pose estimation based on Least-Squares Matching [27], will be fused with torso orientation to improve extraction of focus of attention and to handle cases when the user only rotates the head towards the robot to seek attention. Complementary cues from visual speech detection and facial expression recognition will also be integrated in future versions of the system.

2.2 Linguistic interaction

The linguistic interaction in the system is carried out by components which recognise, understand, and generate embodied natural language. First we describe the components which deal with recognising the user's spoken input: speech recognition and natural language interpretation; we then turn to the components which create the system's spoken output and pass it on to the actuators along with the outputs for the other modalities: natural language generation and the multimodal output generator.

The speech recognition uses the Microsoft Kinect and the associated Microsoft Speech API. We use a grammar in the SRGS format [15] to constrain the recognition and achieve more reliable recognition results, and the output is a list of hypotheses with associated confidence values, along with source localisation information

```
<output>
  <gesture-list>
    <gesture type="Smile"/>
  </gesture-list>
  <action-list>
    <action type="give">
      <object type="drink" name="juice" id="A1"/>
      <person id="id5"/>
    </action>
  </action-list>
  <speech-list>
    <speech type="inform" politeness="4">
      <person id="id5"/>
      <pred type="hand-over">
        <object type="drink" name="juice" id="A1"/>
      </pred>
    </speech>
  </speech-list>
</output>
```

Figure 6: XML for multimodal presentation planning

from the Kinect. To avoid attempting to process the system speech, recognition is not carried out while the robot is speaking. Because we have a relatively constrained grammar, if customers say something which is outside our domain (e.g. "what time is it"), we are likely to get a very low recognition score, and the hypotheses may all be rejected as being below the Speech API's built-in threshold, and therefore not be passed on. In most cases, the top hypothesis is passed to the natural language interpretation module which parses it to provide syntactic and semantic information, using a bi-directional OpenCCG grammar [39] which is also used for the natural language generation. The resulting logical form is passed on to the state management component. The example shown in Figure 5 is produced by the spoken input "Please can I get a Coke?"

On the output side, the multimodal output component receives from the planner an XML representation of the system actions to be carried out by the robot talking head and arms, such as the one in Figure 6. This example would cause the robot to smile while handing a juice to the customer and saying "here is your drink." The verbal output is specified in a format [17] based on Rhetorical Structure Theory (RST) [21] and is used to create a logical form which is then realised by using the OpenCCG grammar mentioned above. The speech output and facial expressions are sent to the talking head, and the gestures to the robot arms.

In subsequent versions of the robot bartender, the coverage of the OpenCCG grammar will be expanded to cover the input and output requirements of the more complex scenarios, and the multimodal output component will also be enhanced to allow more precise timing among the various components of a multimodal output turn.

2.3 State management

The primary role of the state manager is to turn the continuous stream of messages produced by the low-level input and output components into a discrete representation of the world, the robot, and all entities in the scene, combining social, dialogue, and task-based properties. The resulting state is used in two distinct ways in the system processing. On the one hand, the state manager provides a persistent, queryable interface to the state: for example, it stores the world coordinates of all entities as reported by the vision system so that the robot is able to correctly look at a particular agent when needed. On the other hand, it also informs the high-level reasoning components whenever there is a relevant change to the state.

In the current system, the state manager is rule-based. One set of rules infers user social states (e.g., seeking attention) based on the low-level sensor data, using guidelines derived from the study of human-human interactions in the bartender domain [16]: in particular, an agent is considered to be seeking attention if they are close to the bar and oriented towards the bartender. The state manager

also incorporates rules that map from the logical forms produced by the parser into communicative acts (e.g., drink orders), and that use the source localisation from the speech recogniser together with the vision properties to determine which customer is likely to be speaking. A final set of rules determine when new state reports are published, which helps control turn-taking in the system. Details on the representations used in the state manager are given in [30].

In subsequent versions of the system, the state manager will be enhanced to process more complex messages from the updated input and output components, taking into account the associated confidence scores, and also to deal with the more complex state representations that will be required by the updated high-level reasoning system. To address this, we will draw on recent work in social signal processing [35] by training supervised learning classifiers on data gathered from humans interacting with both real and artificial bartenders, using methods similar to those employed by [6, 18].

2.4 High-level planning and monitoring

The high-level planning component uses state reports from the state manager to generate sequences of actions which are sent to the output planner for execution on the robot as speech, head gestures, and arm movements. This component also monitors the execution of such actions, through subsequent state reports, to ensure the system's high-level goals are being met. When failures or plan divergences are detected, actions are replanned as necessary.

To control action selection, we use a knowledge-level planner called PKS (Planning with Knowledge and Sensing) [28, 29] which builds plans in the presence of incomplete information and sensing, by reasoning about how its knowledge state changes due to action. PKS's knowledge state is represented symbolically by a set of five databases, each of which models a particular type of information, interpreted in a modal logic of knowledge. Actions can modify any of the databases in a STRIPS-like [10] manner through additions or deletions which produce changes in the planner's knowledge. To ensure efficient reasoning, PKS restricts the knowledge it can represent while ensuring it is expressive enough to model many types of information that arise in common planning scenarios.

Like other symbolic planners, PKS requires a definition of the actions available to it, initial (knowledge) state, and a goal to be achieved. A plan is successful provided it transforms the initial state to a state where the goals are satisfied. In the bartending scenario, the domain includes definitions for eight parameterised actions, including `greet(?a)` (greet an agent `?a`), `ask-drink(?a)` (ask an agent `?a` for a drink order), `serve(?a,?d)` (serve drink `?d` to agent `?a`), `wait(?a)` (tell agent `?a` to wait), `ack-wait(?a)` (thank agent `?a` for waiting), `ack-thanks(?a)` (respond to an agent `?a` that thanks the bartender), `not-understand(?a)` (alert agent `?a` that their response was not understood), and `bye(?a)` (end an interaction with agent `?a`). These actions are described at an abstract level and include a mix of task, sensory, and linguistic acts.

Unlike many systems that include speech as an input and output modality, we do not use a dedicated dialogue or interaction manager (e.g., TrindiKit [19]). Instead, the planner is a general-purpose problem solving engine, rather than a specialised tool that has been optimised for dialogue. As a result, all actions (linguistic or otherwise) are treated in a similar fashion during plan generation.

The initial state is not hardcoded. Instead, the planner uses the state information passed to it from the state manager. The planner's goal is simply to serve each agent it knows about. This goal is viewed as a rolling target which is reassessed each time it receives a state report from the state manager. For instance, if the appearance of an agent `A1` is reported to the planner as an initial state report, the planner will build a plan of the following form to serve the agent:

`greet(A1)`,	[Greet agent `A1`]
`ask-drink(A1)`,	[Ask `A1` for a drink order]
`serve(A1,drink(A1))`,	[Give the ordered drink to `A1`]
`bye(A1)`.	[Finish the transaction with `A1`]

(`drink(A1)` is a placeholder for the actual drink ordered by `A1`.)

When a plan has been built, it is post-processed by mapping each action into an RST structure (Figure 6) that explicitly encodes the speech, gesture, and robot parts. Actions are then sent to the output planner, one at a time, for eventual execution in the world.

Once action execution has begun, an execution monitor assesses plan correctness, by comparing subsequent state manager reports against the states predicted by the planner. In the case of disagreement, for instance due to unexpected outcomes like action failure, the planner is invoked to construct a new plan using the current state as its initial state. This method is particularly useful for responding to unexpected actions by agents interacting with the bartender.

For example, if the planner receives a report that `A1`'s response to `ask-drink(A1)` was not understood, it will attempt to build a new plan. One possible result is a modified version of the original plan that first informs the agent they were not understood before repeating `ask-drink(A1)` and continuing with the rest of the plan:

`not-understand(A1)`,	[Inform `A1` they were not understood]
`ask-drink(A1)`,	[Ask `A1` for a drink order]
...continue with plan...	

Another useful consequence of this approach is that certain types of over-answering by the interacting agent can be handled by the execution monitor through replanning. For instance, if a `greet(A1)` action by the bartender causes `A1` to respond with a drink order, replanning will construct a new plan that omits the `ask-drink(A1)` action and instead proceeds to serve the drink.

The planner can also deal with multiple agents, as in Figure 2. For instance, with two agents, `A1` and `A2`, one possible plan is:

`wait(A2)`,	[Tell agent `A2` to wait]
`greet(A1)`,	[Greet agent `A1`]
`ask-drink(A1)`,	[Ask `A1` for a drink order]
`serve(A1,drink(A1))`,	[Give the ordered drink to `A1`]
`bye(A1)`,	[Finish the transaction with `A1`]
`ack-wait(A2)`,	[Thank agent `A2` for waiting]
`ask-drink(A2)`,	[Ask `A2` for a drink order]
`serve(A2,drink(A2))`,	[Give the ordered drink to `A2`]
`bye(A2)`.	[Finish the transaction with `A2`]

Agent `A2` is first told to wait and then `A1`'s drink order is taken and the drink is served. After ending `A1`'s transaction, `A2` is thanked for waiting and the drink ordering process is repeated for `A2`.

In the initial version of the bartender domain, the planner only builds simple sequences of actions (i.e., linear plans). In the next version of the system, we will consider plans with contingencies (i.e., plans with branches), in an attempt to construct more robust plans. More details on the bartending domain are provided in [30].

2.5 Robot behaviours

The talking head and the robot arms are controlled by separate software modules. The talking-head controller provides basic behaviours that are called by the output planner when needed. In this study, we used the following behaviours: lip-synchronised speech output of generated sentences, display of emotions (including smiling and frowning), nodding and shaking the head, and turning the head towards an agent or a neutral position. Furthermore, the talking-head controller informs the rest of the system when speech starts and ends and when it executes a facial expression or head gesture. This information is used to stop speech recognition

when the robot was talking to prevent recognition from analysing the robot's utterances, and also to help control turn-taking.

The robot motion planner and simulator components provide a common interface for grasp and place commands from a set of predefined locations. The motion planner generates smooth trajectories and controls the two robot arms in real-time. The simulator shares the same implementation and produces identical trajectories—its single difference lies in the robot hardware abstraction, as it visualises the robot setup and its environment in 3D, allowing the rest of the system to be tested independent of the robot hardware.

Motion planning and robot control make use of the Robotics Library [31, 33]. To create a robust system with deterministic behaviour, we implemented a simple grasping strategy that moves the tool centre point to the desired location with a fixed tool centre orientation. Since the humanoid hands are driven by series elastic actuators and coated with non-slip urethane, we can easily achieve a robust grasp with a simple torque-limited position controller. With this grasping scheme, the robot can grasp various types of bottles of different sizes and materials. For motion planning, locations and via-points are transformed into joint space by the closed-form inverse kinematics given in the Robotics Library. Then, smooth trajectories are generated by quintic polynomial interpolation. For via-points, a tangential velocity is chosen based on the positions of the previous and the next point. Since the number of locations and robot behaviours is limited in the current version, we can enumerate and check all trajectories for collisions with the static environment, checking for collisions at compile time.

In later versions of the bartender, the range of robot actions will be expanded to allow the robot to pick up, put down, and hand over a range of objects at any location; the talking-head actions will also be expanded to cover more complex interaction scenarios.

3. USER EVALUATION

To evaluate the robot bartender, we carried out a user study in which participants enacted variations on the drink-ordering scenario shown in Figure 2 and then answered a short questionnaire regarding their experience of interacting with the bartender. In addition to the questionnaire, we also gathered a range of other measures assessing the quality of the interaction based on data gathered from the system log files. This study serves two purposes: on the one hand, the results provide a useful assessment of the quality of the initial robot bartender system, and one which can serve as a baseline for future evaluations. On the other hand, the study also acts as a formative assessment of the system components, guiding the development of enhanced versions.

3.1 Participants

31 participants (22 male), drawn from university departments outside the robotics group, took part in this experiment. The mean age of the participants was 27.9 (range 21–50), and their mean self-rating of experience with human-robot interaction systems was 2.29 on a scale of 1–5. Neither of these demographic factors had any effect on the study results presented below.

3.2 Scenario

The study took place in a lab, with lighting and background noise controlled as far as possible. Each participant ordered a drink from the robot bartender in the following three scenarios:

1. The participant approached the bartender alone.
2. The participant approached the bartender with a confederate also in view of the cameras but not attempting to attract the bartender's attention.
3. The participant and a confederate both approached the bartender together.

The interactions were similar to that shown in Figure 2. Note that a minimal successful drink-ordering transaction requires three system turns: a greeting from the bartender, serving the drink, and a good-bye from the robot. The transaction could also include any number of requests for the customer's drink order. Each participant was given a list of the possible drinks that could be ordered (water, juice, or Coke), but was not given any further instructions. The robot was static until approached by a customer, and the confederate did not attempt to speak at the same time as the participant. After the three scenarios were completed, the participant completed a short computer-based questionnaire.

3.3 Dependent measures

We gathered two classes of dependent measures: objective measures derived from the system logs and video recordings, and subjective measures gathered from the questionnaire.

3.3.1 Objective measures

The objective measures were divided into three categories, based on those used in the PARADISE framework [37]. To assess **task success** in this scenario, we checked whether customers were correctly detected to be seeking attention, and whether each customer who wanted a drink received one. The **dialogue quality** measures counted the number of attempted user turns that fell below the speech-recognition confidence threshold (see Section 2.2), and the number of timeouts (i.e., moments where the user failed to provide a recognised input when one was expected). Finally, the **dialogue efficiency** measures concentrated on the timing: the time taken for the bartender to acknowledge a customer's bid for attention, the number of system turns, the number of times the robot asked the customer for a drink order, and the duration of the transaction.

3.3.2 Subjective measures

The subjective questionnaire began by asking the participant to rate each of the three interactions on a scale of 1–10, and then asked them to rate the robot and their experience of using it on a number of scales. The survey was based on the GODSPEED questionnaire [5], which is designed to be a standard user measurement tool for human-robot interaction. The survey measured user opinions of the robot on five scales: anthropomorphism (five items), animacy (six items), likeability (six items), perceived intelligence (five items), and perceived safety (three items); the items in the different categories were interleaved. All responses were given on a five-point semantic differential scale, with lower scores corresponding in each case to a more negative assessment of the robot or the interaction.

3.4 Results

3.4.1 Objective results

Table 1a shows the task success results, divided by the scenario. For all 31 participants, the robot detected and attempted to serve the first customer in all scenarios. In scenario 2, the confederate was—correctly—never considered to need attention. In scenario 3, the system detected the second customer in 18 trials, and determined that they wanted attention in 16 trials: the remaining customers were not detected due to a combination of technical vision problems and state-manager rule failure. Overall, of the 109 customers who were determined to need attention, 104 managed to order a drink. All of the unsuccessful transactions were due to technical problems that led to user inputs not being processed.

Scen.	Measure	Count	%
1	Drink served	31/31	100.0
2	Drink served	28/31	90.3
3	Drink #1 served	30/31	96.8
	Drink #2 served	15/16	93.8

(a) Task success

	Mean	Std dev	Min	Max
Low ASR	2.26	1.67	0	6
Timeouts	0.94	2.20	0	11
Response time (ms)	658	2443	9	17920
System turns	5.84	2.87	3	22
Order requests	2.37	2.99	0	20
Duration (s)	49.4	29.6	24.4	164.6

(b) Dialogue quality and efficiency

Table 1: Objective results

Scen.	Mean	Std dev	Min	Max
1	6.65	2.95	2	10
2	6.55	2.79	1	10
3	4.22	3.16	1	10

(a) Scenario quality

Category	Cronbach's α	Mean	Std dev
Anthropomorphism	0.79	2.39	0.75
Animacy	0.83	2.57	0.77
Likeability	0.93	3.73	0.93
Perceived Intelligence	0.84	3.16	0.77
Perceived Safety	0.68	3.56	0.60

(b) GODSPEED questionnaire

Table 2: Subjective results

The results for dialogue quality are shown in the top rows of Table 1b, averaged across all of the 109 drink-ordering transactions. Most transactions included at least one attempted user turn that fell below the ASR confidence threshold, and many also included at least one timeout. Despite this, 74 transactions proceeded with no timeouts; the transactions with a large number of timeouts (≥ 10) were due to input-processing failures as mentioned above.

The bottom rows of Table 1b show the results on the dialogue efficiency measures, again averaged across the 109 transactions. The robot was generally very responsive, often reacting to an attention bid in less than 10 milliseconds; longer response times reflect transactions where the second customer bid for attention while the first drink was being served, and therefore had to wait to be acknowledged. To put these results into context, consider that that a minimal drink-ordering transaction required 3–4 system turns, while the robot arm took 22 seconds to serve a drink. In other words, while many transactions were as efficient as possible, there were also some that took much longer than required.

3.4.2 Subjective results

Table 2a shows the users' judgements of the overall quality of the interaction in the three scenarios, on a scale of 1–10. In general, the participants gave moderately positive ratings to the first two scenarios, and a somewhat more negative rating to the third (two-customer) scenario. There was little correlation among the users' responses to the three judgements ($\alpha = 0.34$).

The overall results from the GODSPEED questionnaire are shown in Table 2b. For each subset of items, we first computed Cronbach's alpha as a measure of internal consistency: as shown in the first column of the table, the consistency for all item categories generally fell into the acceptable range ($\alpha \gtrsim 0.70$). The remaining columns of the table summarise the user responses to each group of questions. The mean responses for anthropomorphism and animacy were around the middle of the five-point semantic differential scale, while the findings for likeability, perceived intelligence, and perceived safety were somewhat above the middle.

Scen.	Function	R^2	Significance
1	$6.65 - 1.54 * \mathcal{N}$(SysTurns)	0.25	SysTurns: $p < 0.01$
2	$6.55 - 1.35 * \mathcal{N}$(SysTurns)	0.21	SysTurns: $p < 0.01$
3	$4.23 + 1.34 * \mathcal{N}$(Serve3-2) $-$ $0.91 * \mathcal{N}$(Duration)	0.10	Serve3-2: $p < 0.05$, Duration: $p \approx 0.14$

Table 3: Predictor functions for scenario quality

3.4.3 Comparing objective and subjective measures

In the preceding sections, we considered a number of objective and subjective measures, all of which varied widely across participants and across trials. We therefore investigated which of the objective measures had the largest effect on users' subjective reactions. Being able to predict subjective user satisfaction from more easily-measured objective properties can be very useful for developers of interactive systems: in addition to making it possible to evaluate systems based on automatically available data without the need for extensive experiments with users, such a performance function can also be used in an online, incremental manner to adapt system behaviour to avoid entering a state that is likely to reduce user satisfaction [20], or can be used as a reward function in a reinforcement-learning scenario [38].

We employed the procedure used in the PARADISE evaluation framework [37] to explore the relationship between the subjective and objective factors. The PARADISE model uses iterative, stepwise multiple linear regression to predict subjective user satisfaction based on objective measures representing the performance dimensions of task success, dialogue quality, and dialogue efficiency, resulting in a predictor function of the following form:

$$\textit{Satisfaction} \quad = \quad \sum_{i=1}^{n} w_i * \mathcal{N}(m_i)$$

The m_i terms represent the value of each measure, while the $\mathcal{N}$ function transforms each measure into a normal distribution using z-score normalisation. Stepwise linear regression produces coefficients (w_i) describing the relative contribution of each predictor to the user satisfaction. If a predictor does not contribute significantly, its w_i value becomes zero after the stepwise process.

We first used the PARADISE procedure to generate predictor functions for the scenario judgements (Table 2a), using the objective results from the relevant scenario as initial factors. The resulting functions are shown in Table 3. The R^2 column indicates the percentage of the variance that is explained by the predictor function, while the *Significance* column gives significance values for each term in the function. The functions for the first two scenarios are very similar: in both cases, the only significant predictor was the number of system turns, which had a negative effect on the judgement, explaining over 20% of the variance. For Scenario 3, the main factors were whether the second customer got served (which had a positive effect), and the duration (which had a negative effect); this function also explains less of the variance (10%).

We then used a similar procedure to generate predictor functions for the responses to the rest of the questionnaire. The objective measures described in Section 3.4.1 were computed on the basis of individual transactions, while each participant provided a single set of responses to the questionnaire; for the purposes of the regression

Category	Function	R^2	Significance
Anthropomorphism	$2.39 - 2.69 * \mathcal{N}$(ServeMean) + $2.20 * \mathcal{N}$(Serve3-1) + $1.96 * \mathcal{N}$(Serve2) + $0.33 * \mathcal{N}$(RespTimeMean)	0.20	RespTimeMean: $p < 0.05$, Serve3-1: $p < 0.05$, Serve2: $p < 0.05$, ServeMean: $p < 0.05$
Animacy	$2.57 - 2.51 * \mathcal{N}$(ServeMean) + $1.94 * \mathcal{N}$(Serve2) + $1.88 * \mathcal{N}$(Serve3-1)	0.09	Serve2: $p < 0.05$, Serve3-1: $p \approx 0.06$, ServeMean: $p \approx 0.07$
Likeability	$3.73 - 3.44 * \mathcal{N}$(ServeMean) + $2.77 * \mathcal{N}$(Serve2) + $2.48 * \mathcal{N}$(Serve3-1)	0.25	Serve2: $p < 0.05$, Serve3-1: $p < 0.05$, ServeMean: $p < 0.05$
Perceived Intelligence	$3.16 - 3.62 * \mathcal{N}$(ServeMean) + $2.75 * \mathcal{N}$(Serve2) + $2.75 * \mathcal{N}$(Serve3-1) + $0.24 * \mathcal{N}$(Serve3-2) − $0.23 * \mathcal{N}$(LowASRMean)	0.29	Serve2: $p < 0.01$, Serve3-1: $p < 0.01$, ServeMean: $p < 0.01$, Serve3-2: $p \approx 0.08$, LowASRMean: $p \approx 0.08$
Perceived Safety	$3.56 - 2.91 * \mathcal{N}$(ServeMean) + $2.21 * \mathcal{N}$(Serve3-1) + $1.94 * \mathcal{N}$(Serve2) + $1.28 * \mathcal{N}$(DurationMean) − $1.15 * \mathcal{N}$(OrderReqMean) − $0.52 * \mathcal{N}$(Serve3-2) − $0.21 * \mathcal{N}$(LowASRMean)	0.31	DurationMean: $p < 0.05$, Serve3-2: $p < 0.05$, Serve3-1: $p < 0.05$, ServeMean: $p < 0.05$, Serve2: $p \approx 0.06$, OrderMean: $p \approx 0.09$, LowASRMean: $p \approx 0.16$

Table 4: Predictor functions for GODSPEED questionnaire categories

analysis, we therefore used the per-participant means as predictors. Table 4 shows the predictor functions that were derived for each of the classes of subjective measures in this study. For most categories, the main factor was the task success: in almost all cases, the scores tended to be higher when drinks were successfully served in Scenarios 2 and 3, with a negative weight on the mean success score compensating for participants for whom both scenarios were successful. Other factors such as the response time and the number of order requests also appear in some functions, but with much lower weights. Most of the predictor functions explain 20-30% of the variance in the questionnaire scores, with the exception of the function for animacy which explains less than 10%.

3.5 Discussion

The overall objective results of this study indicate that the system was generally successful at detecting customers who wanted attention and at serving their drinks. Despite the minimal instructions given to the participants, nearly all of them succeeded in attracting the bartender's attention and ordering a drink. The failures were largely due to easily remedied, low-level technical problems (such as threshold settings or modules behaving improperly) rather than to any higher-level problem with the overall system. While the results for dialogue quality and efficiency suggest that there is room for improvement, it is encouraging that many drink-ordering transactions were successful.

The subjective results are also encouraging: despite the simplicity of the scenario, participants gave the system positive scores for likeability and perceived intelligence, and also gave an overall positive assessment of the interactions in Scenarios 1 and 2. The PARADISE evaluation found that the main contributors to the subjective judgements were task success and dialogue efficiency. The R^2 values for the predictor functions, while in line with those from similar studies [e.g., 13, 20, 38], were generally low, indicating that the users' subjective judgements were also affected by factors other than the objective measures considered here. We are currently analysing the video recordings of the interactions and will use these recordings as the basis for additional measures such as the word error rate from the speech recogniser; we expect that adding such measures will increase the R^2 values.

4. CONCLUSIONS AND FUTURE WORK

We have presented a robot bartender that is designed to work in dynamic, multi-party social situations and have described the architecture and components of the system. We then presented the findings from a user evaluation of the integrated system. The results of this study confirm that users were generally able to order a drink from the bartender in a range of social situations, and also suggest that the main factors influencing the users' subjective opinions were the task success and the dialogue efficiency. These findings provide a baseline for use in follow-up evaluations, and also suggest the areas to focus on for subsequent versions of the system.

Short-term updates to the system include improving the robustness of the input processing to decrease the amount of user speech that is discarded and to increase the number of customers that are detected. We are also currently integrating an alternative high-level decision-making component that uses a policy trained through reinforcement learning to select appropriate system behaviour based on states inferred from the low-level input sensors, using techniques similar to those employed in [11, 32].

In the next user evaluation, we will assess the impact of the social behaviours directly by implementing a version of the system that behaves as in Interaction 1 in Figure 2 and comparing it to the current system. We will also compare the system using the trained RL policy to the current system. A limitation of the current study is that the scenarios were always presented in the same order. The main goal of this study was to test the initial integrated system in a range of conditions, so we do not think that this compromises the overall results; however, we will be sure to counterbalance the scenario order in follow-up studies. Also, in this study participants filled out the full GODSPEED questionnaire series once at the end of all the interactions. In future studies, we will select items from the series that are particularly relevant to the bartender scenario, such as perceived intelligence. Using a shorter questionnaire would also the participants to rate each interaction individually, which would also allow for finer-grained analysis of the results.

In the longer term, we will update all of the components of the system to allow it to support more complex scenarios involving larger numbers of customers in more dynamic scenarios, including customers in groups and dialogues incorporating more of the ordering phenomena found in natural bar interactions [16], such as follow-up questions from the bartender and taking payment. This will involve enhancements to all of the system components as described in Section 2. The updated system will be evaluated in a study similar to this one; the more complex scenarios should also allow the human-robot interactions behaviour to be compared with those found in the human-human data.

5. ACKNOWLEDGEMENTS

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007–2013) under grant agreement no. 270435, JAMES: Joint Action for Multimodal Embodied Social Systems, `http://james-project.eu/`. We thank the ICMI reviewers for their constructive comments, and our JAMES colleagues for useful discussion and collaboration.

6. REFERENCES

[1] A. Argyros and M. Lourakis. 3D tracking of skin-colored regions by a moving stereoscopic observer. *Applied Optics*, 43(2):366–378, Jan. 2004.

[2] W. Bainbridge, J. Hart, E. Kim, and B. Scassellati. The benefits of interactions with physically present robots over video-displayed agents. *International Journal of Social Robotics*, 3:41–52, 2011. doi: 10.1007/s12369-010-0082-7.

[3] H. Baltzakis and A. Argyros. Propagation of pixel hypotheses for multiple objects tracking. In *Proceedings of ISVC 2009*, Nov. 2009.

[4] H. Baltzakis, M. Pateraki, and P. Trahanias. Visual tracking of hands, faces and facial features of multiple persons. *Machine Vision and Applications*, pages 1–17, 2012. doi: 10.1007/s00138-012-0409-5.

[5] C. Bartneck, D. Kulić, E. Croft, and S. Zoghbi. Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. *International Journal of Social Robotics*, 1:71–81, 2009. doi: 10.1007/s12369-008-0001-3.

[6] D. Bohus and E. Horvitz. Dialog in the open world: platform and applications. In *Proceedings of ICMI-MLMI 2009*, pages 31–38, Nov. 2009. doi: 10.1145/1647314.1647323.

[7] C. Breazeal. Socially intelligent robots. *interactions*, 12(2):19–22, 2005. doi: 10.1145/1052438.1052455.

[8] G. Castellano, I. Leite, A. Pereira, C. Martinho, A. Paiva, and P. W. McOwan. Affect recognition for interactive companions: challenges and design in real world scenarios. *Journal on Multimodal User Interfaces*, 3(1):89–98, 2010. doi: 10.1007/s12193-009-0033-5.

[9] K. Dautenhahn. Socially intelligent robots: dimensions of human-robot interaction. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1480):679–704, 2007. doi: 10.1098/rstb.2006.2004.

[10] R. E. Fikes and N. J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2:189–208, 1971. doi: 10.1016/0004-3702(71)90010-5.

[11] M. Frampton and O. Lemon. Recent research advances in reinforcement learning in spoken dialogue systems. *The Knowledge Engineering Review*, 24(4):375–408, 2009. doi: 10.1017/S0269888909990166.

[12] S. S. Ge and M. J. Matarić. Preface. *International Journal of Social Robotics*, 1(1):1–2, 2009. doi: 10.1007/s12369-008-0010-2.

[13] M. Giuliani, M. E. Foster, A. Isard, C. Matheson, J. Oberlander, and A. Knoll. Situated reference in a hybrid human-robot interaction system. In *Proceedings of INLG 2010*, 2010.

[14] T. Horf, R. Roller, and S. Wilske. MiCo: The robotic bartender for mini-cocktails. `http://www.coli.uni-saarland.de/courses/lego-04/page.php?id=barkeeper`, 2004.

[15] A. Hunt and S. McGlashan. Speech recognition grammar specification version 1.0. W3C recommendation, W3C, Mar. 2004. http://www.w3.org/TR/2004/REC-speech-grammar-20040316/.

[16] K. Huth. Wie man ein Bier bestellt. Master's thesis, Universität Bielefeld, 2011.

[17] A. Isard and C. Matheson. Rhetorical structure for natural language generation in dialogue. In *Proceedings of SemDial 2012*, 2012.

[18] A. Kapoor, W. Burleson, and R. W. Picard. Automatic prediction of frustration. *International Journal of Human-Computer Studies*, 65(8):724–736, 2007. doi: 10.1016/j.ijhcs.2007.02.003.

[19] S. Larsson and D. Traum. Information state and dialogue management in the TRINDI dialogue move engine toolkit. *Natural Language Engineering*, 6(3&4):323–340, 2000. doi: 10.1017/S1351324900002539.

[20] D. J. Litman and S. Pan. Designing and evaluating an adaptive spoken dialogue system. *User Modeling and User-Adapted Interaction*, 12 (2–3):111–137, 2002.

[21] W. C. Mann and S. A. Thompson. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281, 1988.

[22] E. Márquez Segura, M. Kriegel, R. Aylett, A. Deshmukh, and H. Cramer. How do you like me in this: User embodiment preferences for companion agents. In *Proceedings of IVA 2012*, Sept. 2012.

[23] T. Masuda and D. Misaki. Development of Japanese green tea serving robot "T-Bartender". In *Proceedings of ICMA 2005*, volume 2, pages 1069–1074, July 2005. doi: 10.1109/ICMA.2005.1626700.

[24] Y. Matsusaka, T. Tojo, and T. Kobayashi. Conversation robot participating in group conversation. *IEICE Transactions on Information and Systems*, 86(1):26–36, 2003.

[25] B. Mutlu, T. Shiwa, T. Kanda, H. Ishiguro, and N. Hagita. Footing in human-robot conversations: how robots might shape participant roles using gaze cues. In *Proceedings of HRI 2009*, pages 61–68, 2009. doi: 10.1145/1514095.1514109.

[26] M. Pateraki, H. Baltzakis, and T. P. Visual tracking of hands, faces and facial features as a basis for human-robot communication. In *Proceedings of the IROS 2011 Workshop on Visual Tracking and Omnidirectional Vision*, September 2011.

[27] M. Pateraki, H. Baltzakis, and P. Trahanias. Using Dempster's rule of combination to robustly estimate pointed targets. In *Proceedings of ICRA 2012*, May 2012.

[28] R. P. A. Petrick and F. Bacchus. A knowledge-based approach to planning with incomplete information and sensing. In *Proceedings of AIPS-2002*, pages 212–221, Apr. 2002.

[29] R. P. A. Petrick and F. Bacchus. Extending the knowledge-based approach to planning with incomplete information and sensing. In *Proceedings of KR-2004*, pages 613–622, June 2004.

[30] R. P. A. Petrick and M. E. Foster. What would you like to drink? recognising and planning with social states in a robot bartender domain. In *Proceedings of CogRob 2012*, 2012.

[31] M. Rickert. *Efficient Motion Planning for Intuitive Task Execution in Modular Manipulation Systems*. Dissertation, Technische Universität München, 2011.

[32] V. Rieser and O. Lemon. *Reinforcement Learning for Adaptive Dialogue Systems: A Data-driven Methodology for Dialogue Management and Natural Language Generation*. Springer, 2011. doi: 10.1007/978-3-642-24942-6.

[33] Robotics Library. URL `http://roblib.sf.net/`.

[34] M. Sigalas, H. Baltzakis, and P. Trahanias. Visual tracking of independently moving body and arms. In *Proceedings of IROS '09*, Oct. 2009.

[35] A. Vinciarelli, M. Pantic, and H. Bourlard. Social signal processing: Survey of an emerging domain. *Image and Vision Computing*, 27(12):1743–1759, 2009. doi: 10.1016/j.imavis.2008.11.007.

[36] J. Wainer, D. J. Feil-Seifer, D. A. Shell, and M. J. Matarić. Embodiment and human-robot interaction: A task-based perspective. In *Proceedings of IEEE RO-MAN 2007*, pages 872–877, Aug. 2007. doi: 10.1109/ROMAN.2007.4415207.

[37] M. Walker, C. Kamm, and D. Litman. Towards developing general models of usability with PARADISE. *Natural Language Engineering*, 6(3&4):363–377, 2000. doi: 10.1017/S1351324900002503.

[38] M. A. Walker. An application of reinforcement learning to dialogue strategy selection in a spoken dialogue system for email. *Journal of Artificial Intelligence Research*, 12:387–416, 2000.

[39] M. White. Efficient realization of coordinate structures in Combinatory Categorial Grammar. *Research on Language and Computation*, 4(1):39–75, 2006. doi: 10.1007/s11168-006-9010-2.