Overview

1. Motivation for social interaction and robotics
2. Overview of the JAMES project
3. Design of the JAMES system
4. What does JAMES do (and why does it do it)?
5. Interacting with JAMES

⇒ Focus on design, implementation, and operation.

Two people walk into another bar...

Two people, A and B, each individually approach a bartender.

Bartender (to A): How can I help you?
Person A: A pint of cider, please.

Person C approaches the bartender and attracts his attention by gesturing.

Bartender (to C): How can I help you?
Person C: I’d like a pint of bitter.
Bartender: (Serves C)

Bartender (to B): What will you have?
Person B: A glass of red wine.
Bartender: (Serves B)
Bartender: (Serves A)
Two interactions

Two people, A and B, each individually approach a bartender

Bartender (to A): How can I help you?
Person A: A pint of cider, please.

Person C approaches the bartender and attracts his attention by gesturing

Bartender (to C): How can I help you?
Person C: I’d like a pint of bitter.

Bartender (to B): What will you have?
Person B: A pint of Guinness.

Bartender: (Serves A)
Bartender: (Serves C)
Bartender: (Serves B)
Bartender: (Serves A)

Bartender (to C): Thanks for waiting.
Person C: I’d like a pint of bitter.
Bartender: (Serves C)

Two interactions (2)

Two people, A and B, each individually approach a bartender

Bartender (to A): How can I help you?
Person A: A pint of cider, please.

Person C approaches the bartender and attracts his attention by gesturing

Bartender (to C): How can I help you?
Person C: I’d like a pint of bitter.

Bartender: (Serves A)
Bartender: (Serves C)
Bartender: (Serves B)
Bartender: (Serves A)

Bartender (to B): What will you have?
Person B: A pint of Guinness.

Bartender: (Serves B)
Bartender: (Serves C)
Bartender: (Serves A)

Bartender (to C): Thanks for waiting.
Person C: I’d like a pint of bitter.
Bartender: (Serves C)

• Is one interaction “better” than the other?
• Does the purpose of the interaction matter?
• What if some (or all) of the participants are robots?

Why social interaction?

• Successful task interaction often relies on social interaction.
  – May be several ways to achieve a task-based goal.
  – Appropriate social behaviour can lead to higher participant satisfaction.

• Social interaction can be seen as an instance of joint action.
  – Involves coordination of participant actions.
  – Inherently multimodal: speech, gesture, gaze, expression, etc.

• Social interaction is often multi-party, dynamic, short-horizon.
  – In contrast to one-on-one, companion-style relationships.
  – Interactions are often ‘one shot’; may not have an opportunity to recover from a poor interaction.

Some questions to consider...

• Why should we care about appropriate social behaviour?
• Why should we care about designing a robot that is capable of such interactions?
• What impact does social interaction have on the design of a robot?
• Does the design of a robot impact the types of interactions (social or otherwise) that it can produce?
• Does appropriate social behaviour really have an impact on the interactions humans have with robots?
  • ...?
Meet the bartender: JAMES

Objectives

1. **Data collection and analysis**: Record and analyse the social and task-based behaviour of humans engaged in multimodal joint activities.

2. **Social modelling**: Design and train a model of social interaction, using annotated data from the human experiments.

3. **Representation and learning**: Endow the model with the ability to learn and adapt to human behaviours, and handle partial or uncertain information about the physical world and mental states of human users.

4. **Implementation**: Implement the model of social interaction on a physical robot platform, initially based on the JAST framework.

5. **Evaluation**: Evaluate the implemented robot system with multiple human users in a bartender scenario.

Target scenario: task-based social interaction

- Robot bartender must respond to user requests in a dynamic setting with multiple users and short interactions in German or English.
- Interactions incorporate both task-based aspects (e.g., ordering and serving drinks) and social aspects (e.g., managing multiple interactions).
- How important is social interaction?
Research themes in JAMES

• How should we design our robot in order to achieve the project’s objectives, given our requirements and constraints?

Design considerations: what do humans do?

• Study how human customers interact with human bartenders to identify the (non-verbal) signals humans are using. How can these results be applied to the robot bartender?

Slate video

• What activities are supported?
  – Asking customers for drinks
  – Clarifying drink requests
  – Handing over drinks
  – Keeping track of the order people arrive at the bar
  – Identifying groups
  – ...

• What activities are not supported?
  – Physically pouring drinks
  – Handling money
  – Small talk
  – ...

**Design considerations: hardware and software**

- Make use of existing and available robot hardware as much as possible
  - Cat head
  - Industrial arms
  - Lab infrastructure
- Supplement existing hardware
  - Kinect
  - 3D cameras
  - New robot hands
  - Tablets
- Use modern software development tools
  - Module-dependent choice of programming language: Java, Python, C++, C, ...
  - Internet Communications Engine (ICE) for communication between components.
  - Modern build system (cmake)
  - Redistributable packages across a range of development and build environments (Linux, Mac, Windows).

**Design considerations: continual development**

**Design considerations: other factors**

- Distributed design
  - Global agreement about the aims of the system.
  - No global control over the internal design of individual components/modules.
  - Individual components in the system are the responsibility of particular partners.
  - Interfaces are agreed upon in a pairwise fashion between individual components.
- Practical considerations
  - Budgets are tight
  - Time is tight
  - Research agendas are (somewhat) fixed
- Geographical considerations
  - The development team was spread over Europe
  - The robot was in Munich and isn't very portable...
  - Testing? Debugging? Evaluation?

**JAMES system architecture**
### Simplified architecture

- **State Manager**
- **Output Planner**
- **Parser**
- **Speech Recogniser**
- **Talking-Head Controller**
- **Robot Motion Planner**

#### Real World
- **Visual Processor**
- **Planner/Execution Monitor**

### Hardware and simulation

- **Physical hardware (shown in video)**
- **Simulated hardware (used for demo)**

**Credit:** A. Gaschler and M. Giulioni, fortiss GmbH

- Provides the robot's embodiment and supplies the primary means for the robot to interact with the real world.

### Visual processing

- **Full system (shown in video)**
- **Kinect-only (used for demo)**

**Credit:** M. Patrakl and M. Sigalas, FORTH

- Provides visual information about the customers in a scene, including their location and body posture.

### Speech recognition and parsing

- **Parsed speech input**
  - *What would you like?*
  - *A water*

- **State update**
  - `add(Kf.request(A1)=water)`

**Credit:** S. Petrick / Case Studies in Design Informatics 1 / The JAMES robot bartender: design and operation / 2015-10-08
**State management**

- Provides an abstract description of what the robot currently believes about the world and the interaction.

**Planning and reasoning**

- Decides what action the robot should perform in the current context.

**Multimodal output planning**

- Processes a high-level action into lower-level actions that control the cat head (speech, head posture) and robot arms.

**Software interface**

- Monitors and controls all components of the system.
**Target scenario: revisited**

Two people, A and B, each individually approach a bartender.

Bartender (to A):
How can I help you?

Person A:
A pint of cider, please.

Person C approaches the bartender and attracts his attention.

Bartender (nods at A, then to C):
Just a moment please.

Bartender:
(Serves A)

Bartender (to B):
What will you have?

Person B:
A glass of red wine.

Bartender (nods at B):
(Serves B)

Bartender (to C):
Thanks for waiting.

Person C:
I’d like a pint of bitter.

Bartender (nods at C):
(Serves C)

- **Challenge:** customers should be able to interact with JAMES in a manner similar to how they would interact with a human bartender.

**Robot bartender actions**

- **Available actions**
  - `greet(?a)`
  - `ask-drink(?a)`
  - `serve(?a, ?d)`
  - `wait(?a)`
  - `ack-order(?a)`
  - `not-understand(?a)`
  - `bye(?a)`

- Inspired by data collected from customers in the human studies.
- The choice of which action the robot should perform is determined by using artificial intelligence automated planning techniques.

**Automated planning actions**

- **Action:** `ask-drink(?a : agent)`
  - **Preconditions:** \( K(\text{inTrans} = ?a) \& !K(\text{ordered}(?a)) \& !K(\text{otherAttnReq}) \& !K(\text{badASR}(?a)) \)
  - **Effects:** \( \text{add}(Kf, \text{ordered}(?a)) \), \( \text{add}(Kv, \text{request}(?a)) \)

- **Action:** `serve(?a : agent, ?d : drink)`
  - **Preconditions:** \( K(\text{inTrans} = ?a) \& K(\text{ordered}(?a)) \& Kv(\text{request}(?a)) \& K(\text{request}(?a) = ?d) \& !K(\text{otherAttnReq}) \& !K(\text{badASR}(?a)) \& K(\text{ackOrder}(?a)) \)
  - **Effects:** \( \text{add}(Kf, \text{served}(?a)) \)

**A simple interaction**

- `greet(A1)`, Greet customer A1
- `ask-drink(A1)`, Ask A1 for drink order
- `ack-order(A1)`, Acknowledge A1’s order
- `serve(A1, request(A1))`, Give the drink to A1
- `bye(A1)`, End the transaction

- Simplest possible standard interaction in the single customer case.
Interacting with two customers

wait(A2),
greet(A1),
ask-drink(A1),
ack-order(A1),
serve(A1, request(A1)),
bye(A1),
ack-wait(A2),
ask-drink(A2),
ack-order(A2),
serve(A2, request(A2)),
bye(A2).

Tell customer A2 to wait
Greet customer A1
Ask A1 for drink order
Acknowledge A1’s order
Give the drink to A1
End A1’s transaction
Thank A2 for waiting
Ask A2 for drink order
Acknowledge A2’s order
Give the drink to A2
End A2’s transaction

- Each customer’s interaction is similar to the single customer case with additional actions to manage the interaction order.

Recovering when things go wrong

- Low-confidence speech recognition / timeouts
  ask-drink(A1)
  ???
  [Replan]
  not-understand(A1)
  ask-drink(A1)
  ...
  [Continue with old plan]

- Overanswering
  greet(A1)
  ???
  [Replan]
  serve(A1, request(A1))
  bye(A1)
  ...

- Interactions often lead to unexpected outcomes. A mechanism is needed to recover from such situations. Replanning is used in JAMES.

A more complex interaction

wait(A3, G1)
greet(A1, G1)
ask-drink(A1, G1)
ack-order(A1, G1)
ask-drink(A2, G1)
ack-order(A2, G1)
serve(A1, request(A1, G1))
serve(A2, request(A2, G2))
bye(A2, G1)
ack-wait(A3, G2)
ask-drink(A3, G2)
ack-order(A3, G2)
serve(A3, request(A3, G3))
bye(A3, G2)

Tell G2 to wait (with a nod)
Greet group G1
Ask A1 for drink order
Acknowledge A1’s order
Ask A2 for drink order
Acknowledge A2’s order
Give the drink to A1
Give the drink to A2
End G1’s transaction
Acknowledge G2’s wait
Ask A3 for drink order
Acknowledge A3’s order
Give the drink to A3
End G2’s transaction

JAMES interaction video

http://youtu.be/8k7Pd-CbbgE
http://james-project.eu/
**Experimental results**

- Action selection time is typically quite short (<0.1s), which doesn’t impact the system’s reaction time.
  - Less than 2s is usually okay.
  - Robot motions are slow.
  - Frequent replanning to recover from problems in the interaction.

- **Study 1**: system tested with 2 customers at a time in a drink ordering scenario (31 participants × 3 interactions each), 95% success rate on delivering correct drinks.

- **Study 2**: more complex scenario (3 customers at a time, 40 participants), group detector, task only vs. social domain, 87% success rate.

**Measure**

<table>
<thead>
<tr>
<th>Measure</th>
<th>Mean</th>
<th>Median</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time to first drink</td>
<td>3.63</td>
<td>3.15</td>
<td>2</td>
<td>5.5</td>
</tr>
<tr>
<td>Total system time</td>
<td>11.08</td>
<td>10.56</td>
<td>9.8</td>
<td>12.5</td>
</tr>
<tr>
<td>Task-only</td>
<td>5.5</td>
<td>5.15</td>
<td>5</td>
<td>6.5</td>
</tr>
<tr>
<td>Fall social (sd)</td>
<td>15.5</td>
<td>15.5</td>
<td>15</td>
<td>16</td>
</tr>
</tbody>
</table>

**Category**

- Anticipation: 2.75 (0.75), 2.62 (0.68), -0.75 (0.99)
- German: 2.75 (0.75), 2.62 (0.68), -0.75 (0.99)
- English: 3.53 (0.94), 3.28 (0.85), -0.73 (0.75)
- Attention: 3.68 (0.98), 3.26 (0.86), -0.75 (0.99)
- German: 3.68 (0.98), 3.26 (0.86), -0.75 (0.99)
- English: 3.53 (0.94), 3.28 (0.85), -0.73 (0.75)
- Likelihood: 4.45 (1.01), 3.78 (1.07), -0.54 (1.36)
- German: 4.45 (1.01), 3.78 (1.07), -0.54 (1.36)
- English: 4.76 (0.92), 3.94 (1.27), -0.78 (0.86)
- Perf. Intelligence: 3.77 (0.94), 3.27 (0.85), -0.54 (0.86)
- German: 3.77 (0.94), 3.27 (0.85), -0.54 (0.86)
- English: 4.22 (0.85), 3.79 (0.75), -0.44 (0.94)
- Perf. Safety: 4.19 (1.01), 3.971 (1.07), -0.19 (1.20)
- German: 4.19 (1.01), 3.971 (1.07), -0.19 (1.20)
- English: 4.23 (1.01), 3.971 (1.07), -0.19 (1.20)

Ron Petrick / Case Studies in Design Informatics 1 / The JAMES robot bartender: design and operation / 2015-10-08

**References**


For more information on the JAMES Project visit [http://james-project.eu/](http://james-project.eu/).