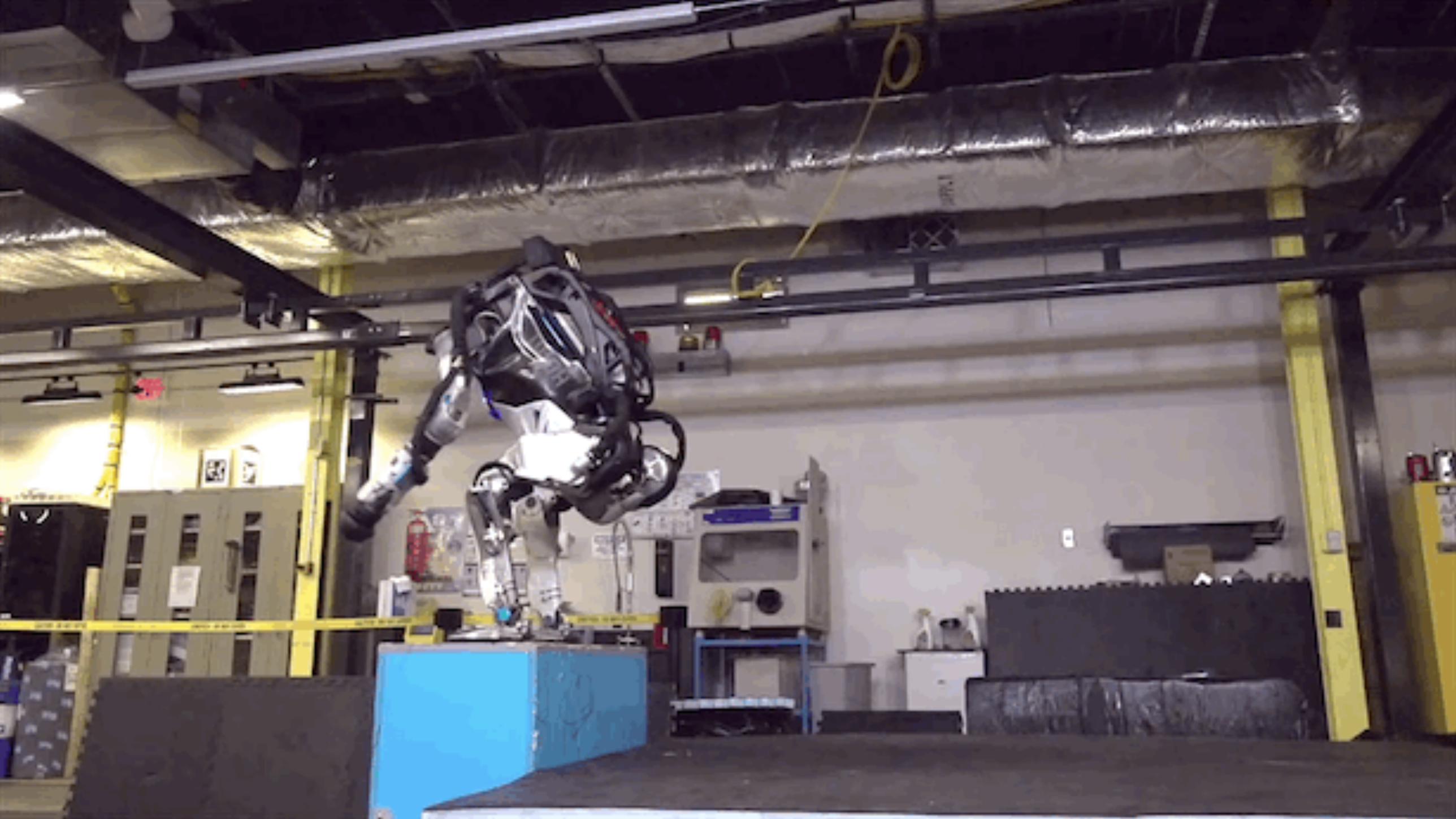
# Group-based Emotions in Teams of Humans and Robots

Filipa Correia, Samuel Mascarenhas, Rui Prada, Francisco S. Melo & Ana Paiva

# What are Group-based Emotions?

FENWAY PARK
Microsoft



No attribution of membership

Individual-based Emotions

**Attribution of membership** to that social group

+

**Event** is relevant for a social group

↓

Group-based Emotions

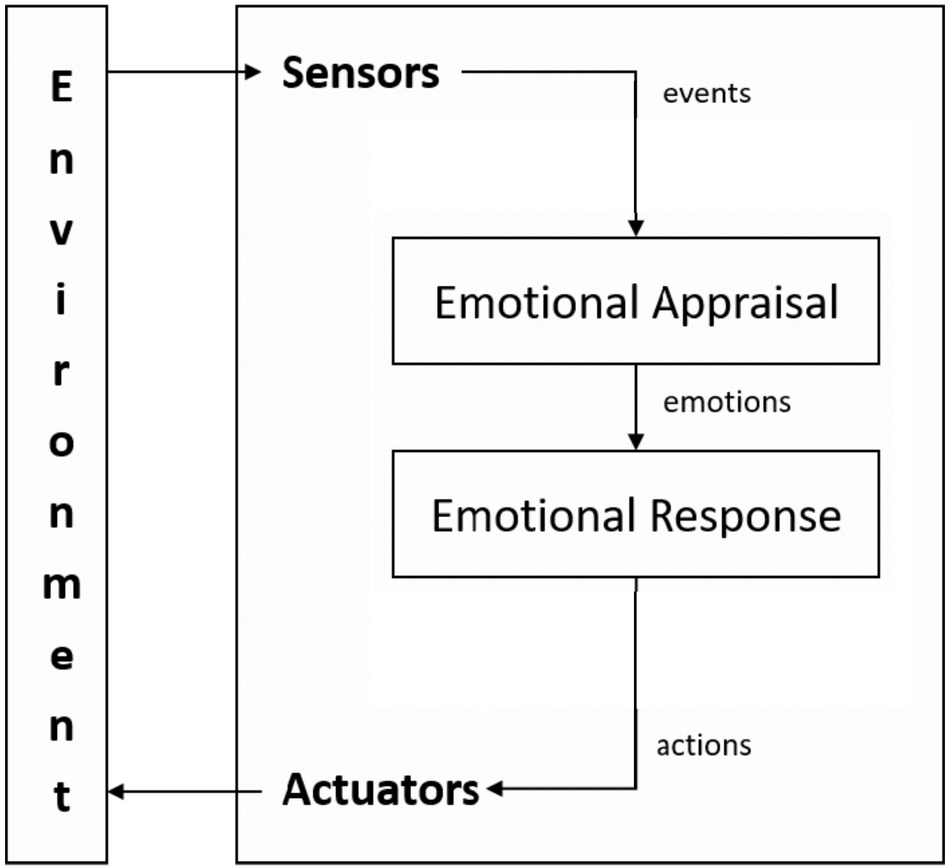
# Motivation to analyse GbE in HRI

# Motivation

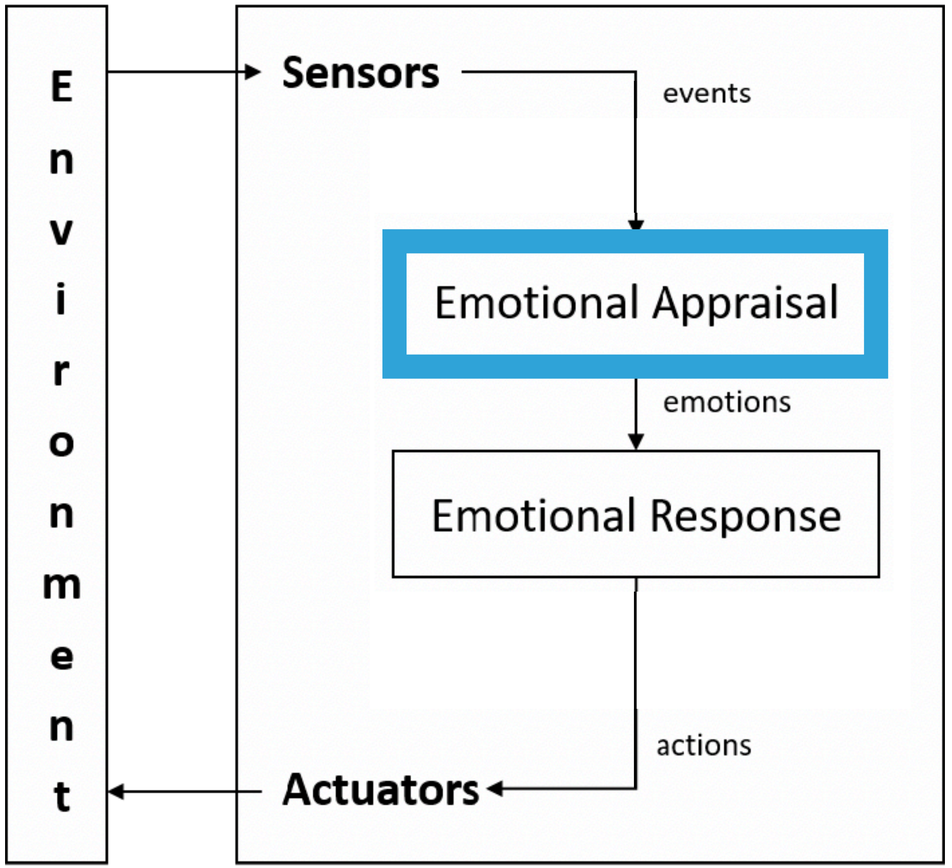
- Intergroup interactions
- Cohesion of the social group
- Trust and Group Identification may lead to positive team performance

# Current models for generation of emotions

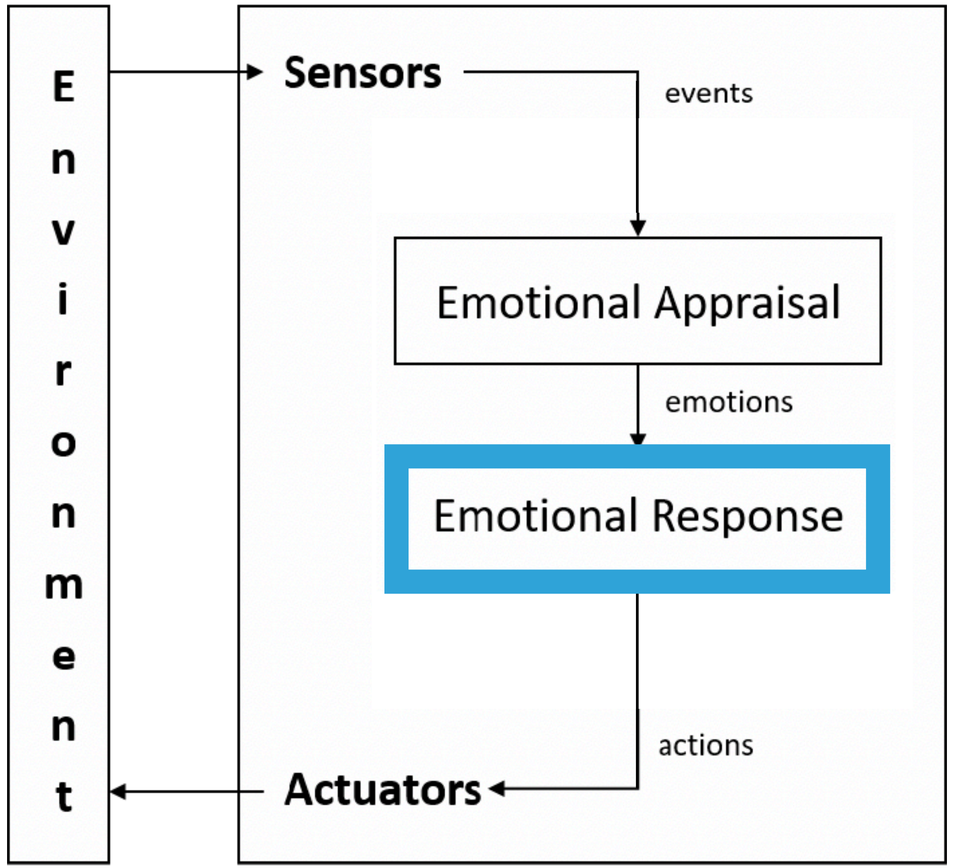
# Current models for generation of emotions



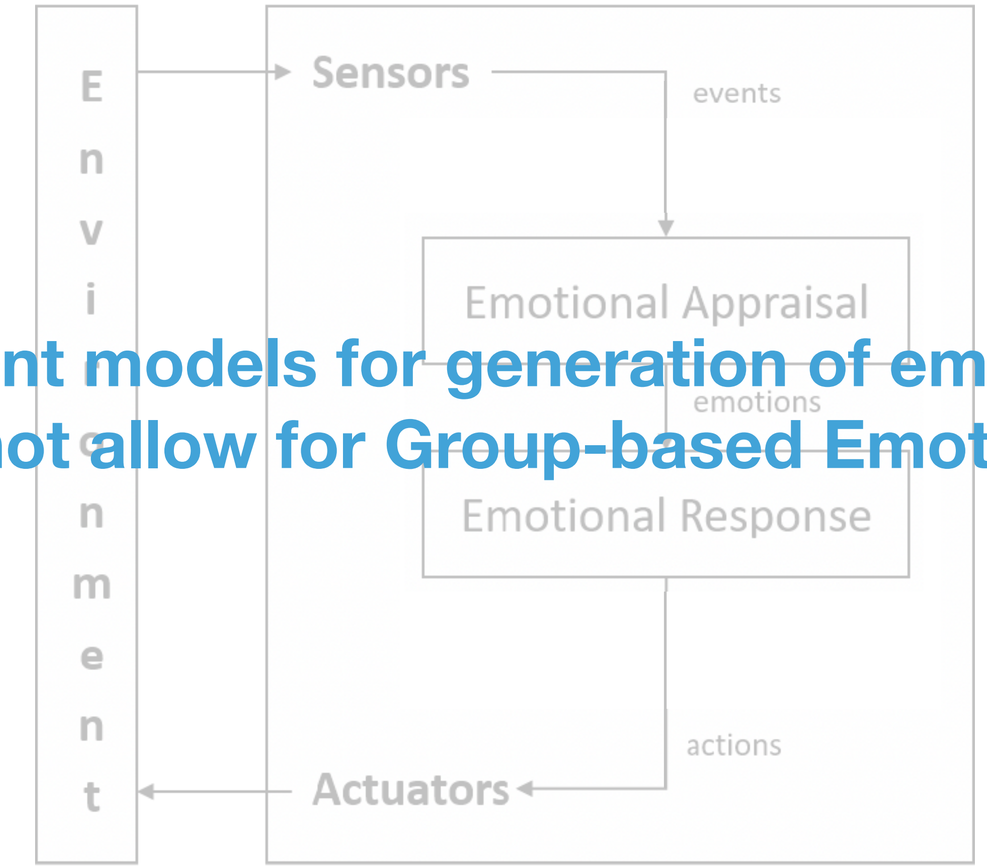
# Current models for generation of emotions



# Current models for generation of emotions



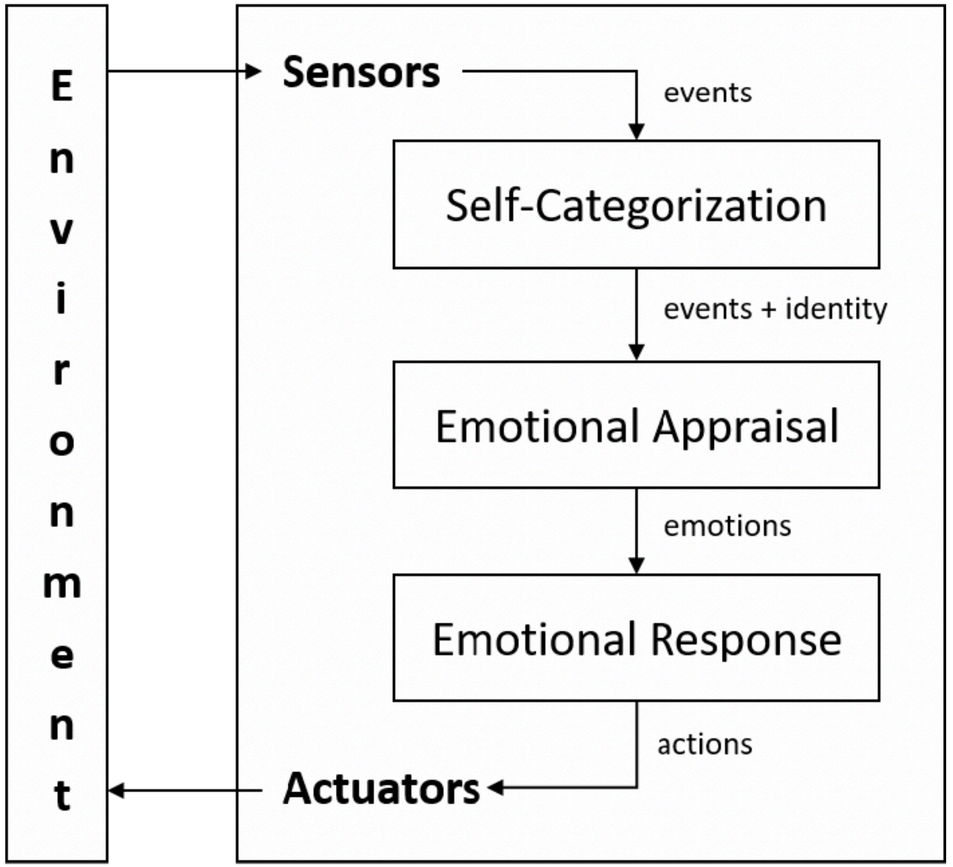
# Current models for generation of emotions




**Current models for generation of emotions do not allow for Group-based Emotions!**

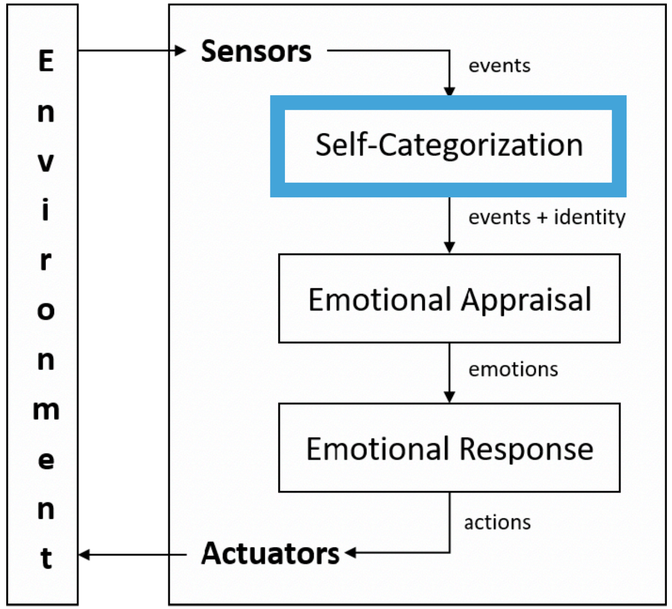
# A model for GbE in social robotic characters

Based on the psychological model of GbE [1]

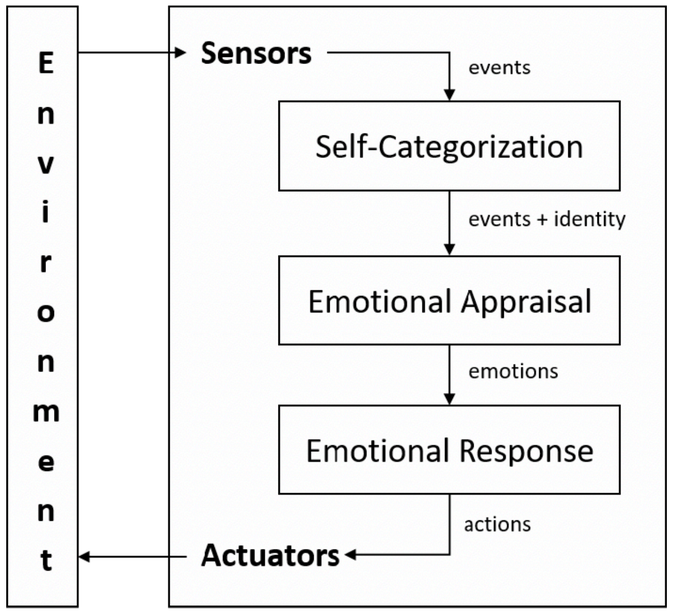



[1] **Goldenberg**, Amit, et al. "**The process model of group-based emotion: Integrating intergroup emotion and emotion regulation perspectives**." *Personality and Social Psychology Review* 20.2 (2016): 118-141.

# A model for GbE in social robotic characters

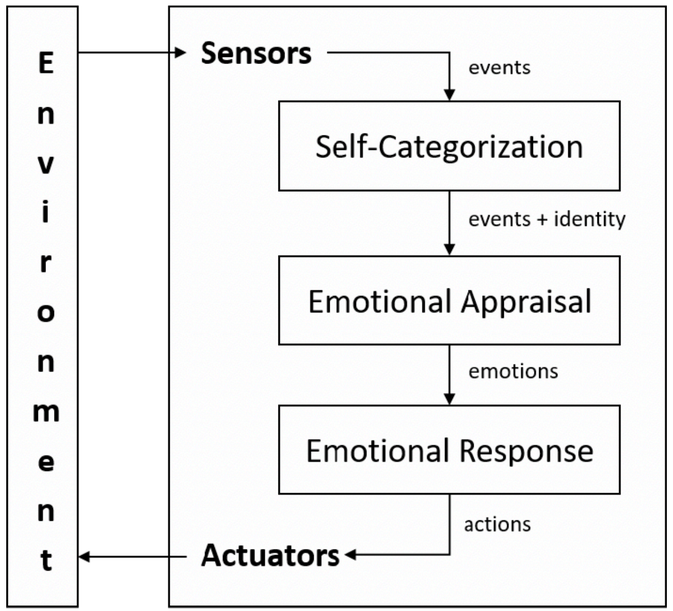


# A model for GbE in social robotic characters



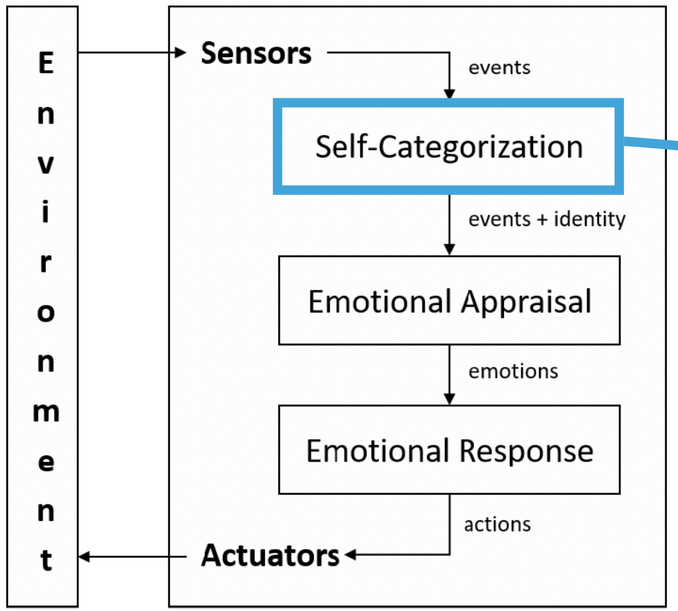

```
while true do
    self ← Robot.Name
    e ← Sensors.PerceiveNewEvent()
    SG ← ContextManager.GetSalientSocialGroups()
    if SG ≠ ∅ then
        g ← IdentityManager.SelfCategorisation(SG, self)
        if e.ResponsibleAgent ∈ g then
            e.ResponsibleAgent ← g.Name
            self ← g.Name
        end if
    end if
    AV ← Appraisal.DetermineVariables(e)
    E ← Appraisal.GenerateEmotions(AV, self)
    se ← StrongestEmotion(E)
    for all c ∈ Actuators.GetEmotionChannels() do
        Express(se, c)
    end for
end while
```

# A model for GbE in social robotic characters



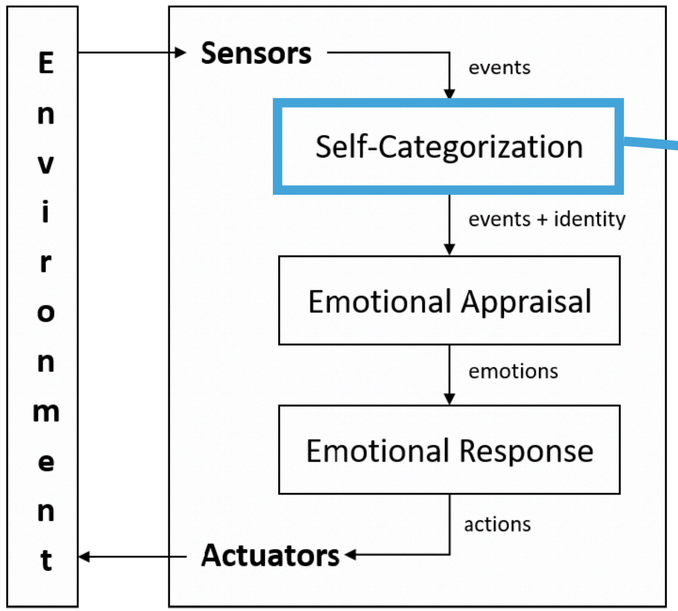

```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

# A model for GbE in social robotic characters



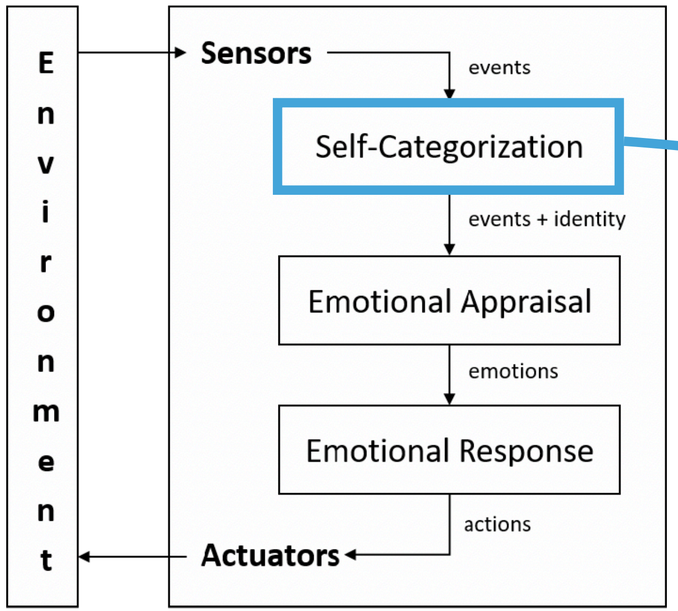

```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

# A model for GbE in social robotic characters



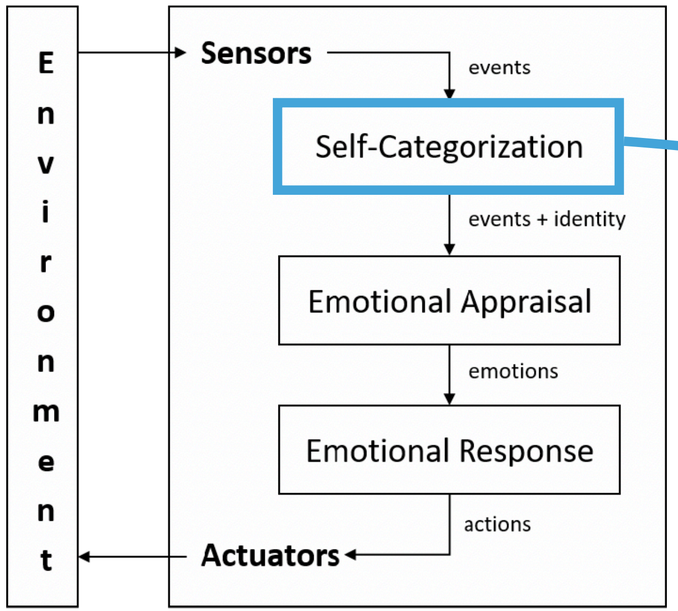

```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

# A model for GbE in social robotic characters



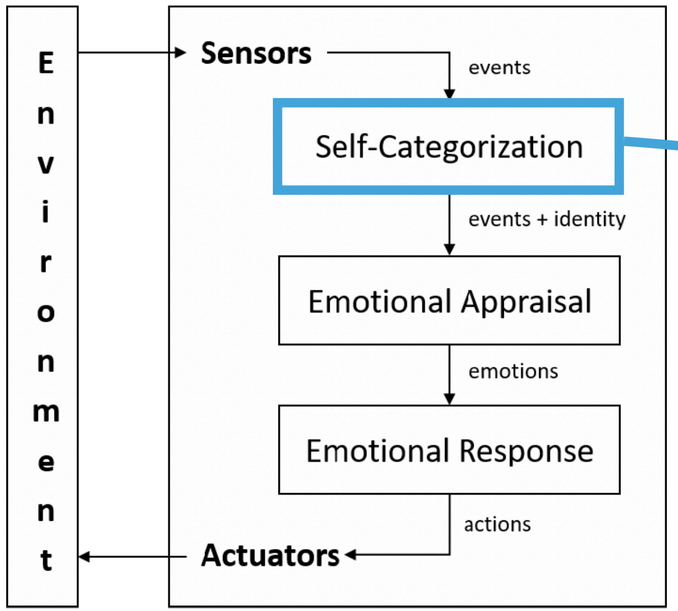

```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

# A model for GbE in social robotic characters



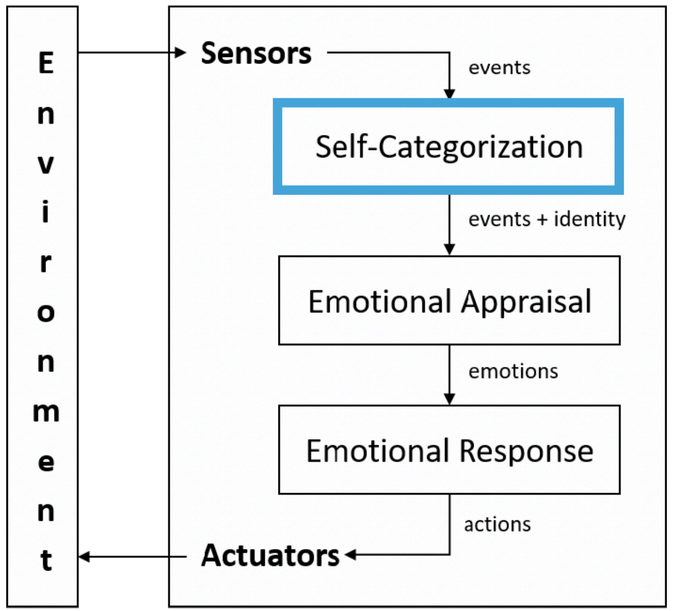

```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

# A model for GbE in social robotic characters



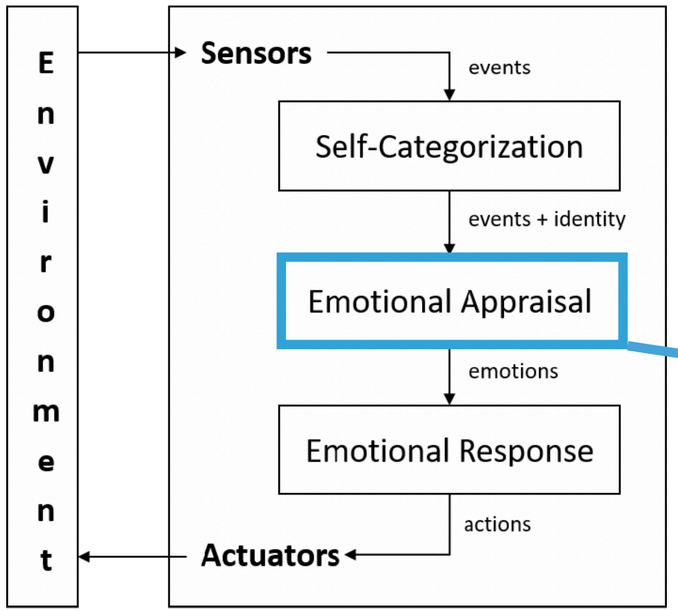

```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

# A model for GbE in social robotic characters



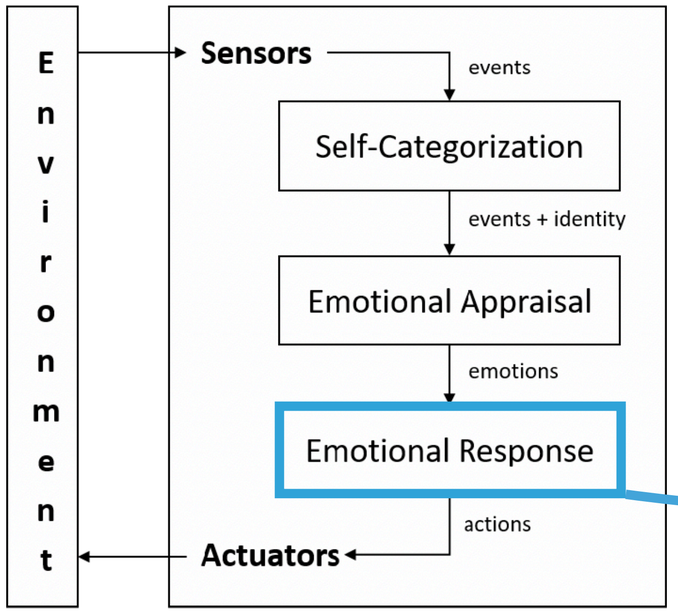

```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

# A model for GbE in social robotic characters




```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

# A model for GbE in social robotic characters

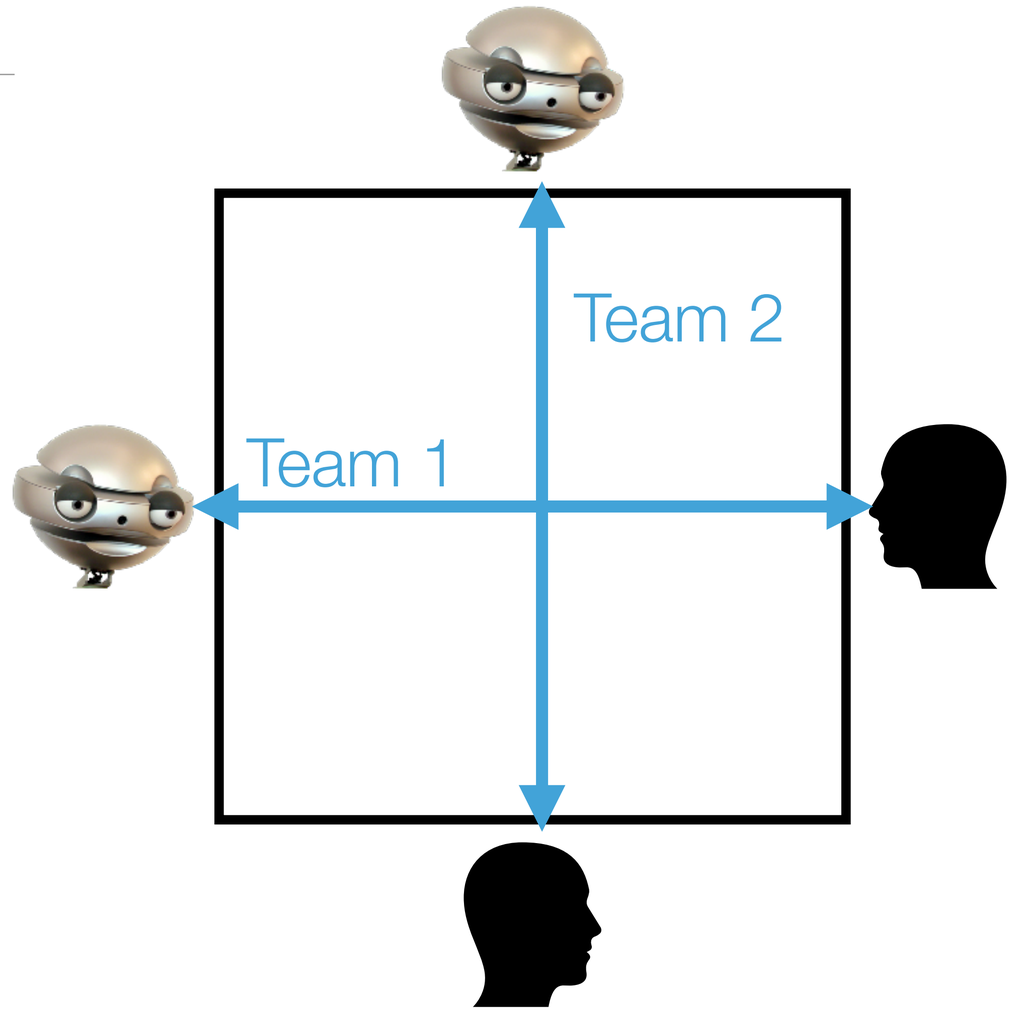



```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

# Scenario

# Card game scenario

- Trick-taking card game
- 2 adversarial teams
- Winning team is the one with more points
  - In-group
  - Out-group

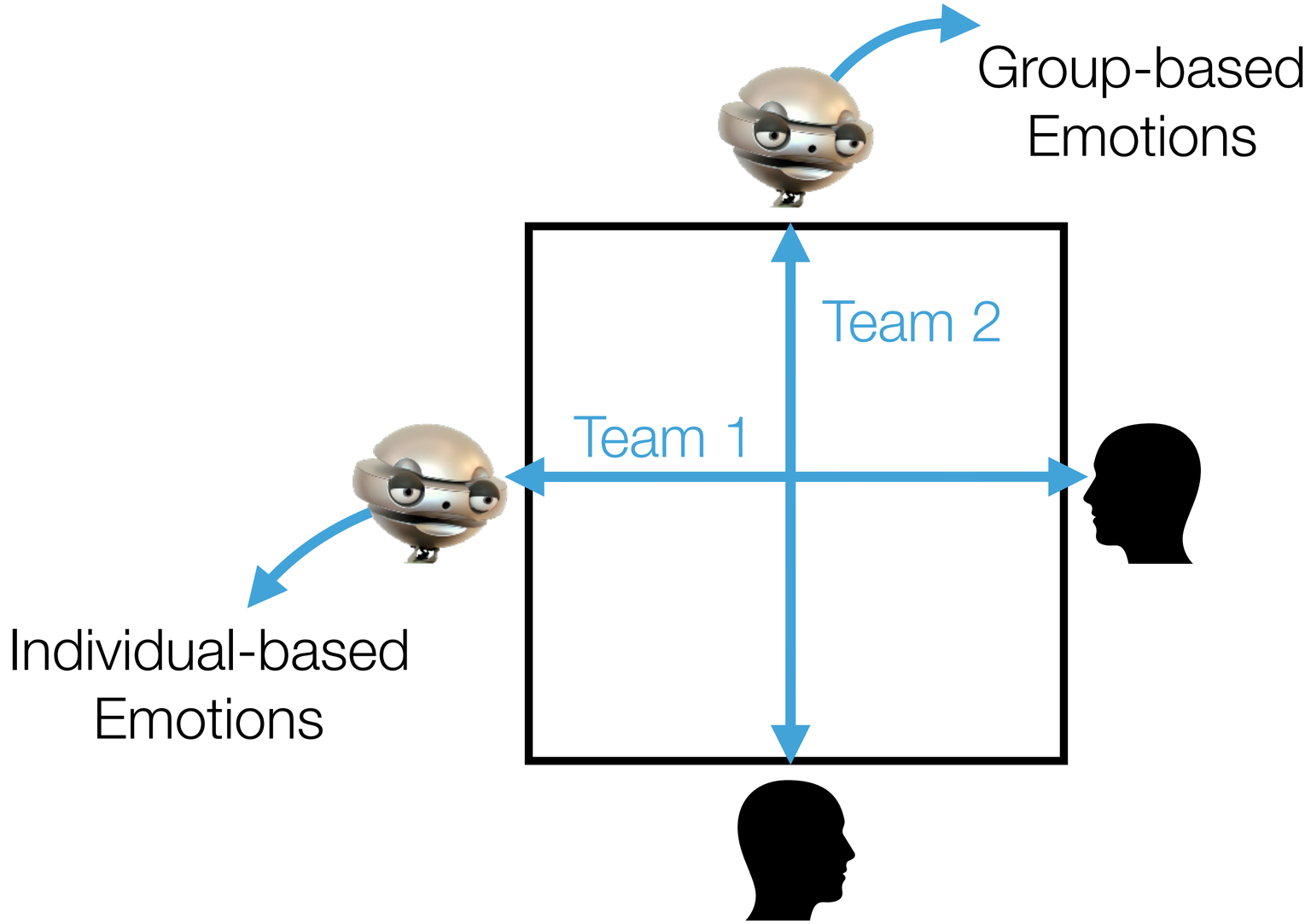


# Hypotheses

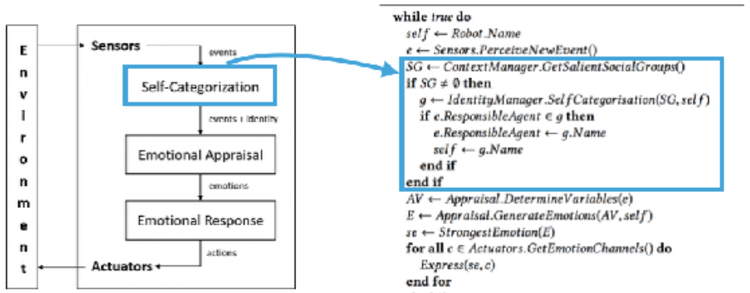
# Hypotheses

- **H1**: Participants will have a **stronger Group Identification** with a robotic partner that expresses GbE.
- **H2**: Participants will have a more **positive perception** of a robotic partner that expresses GbE.
- **H3**: Participants will have a **higher degree of Group Trust** with a robotic partner that expresses GbE.

# Applying the model

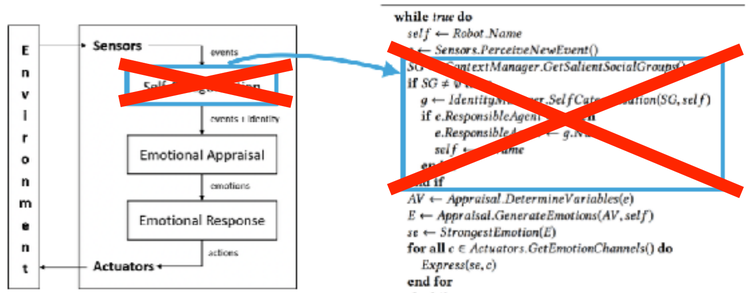

Group-based
Emotions
Team 2
Team 1
Individual-based
Emotions

# How?




```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

Group-based Emotions




```
while true do
  self ← Robot.Name
  e ← Sensors.PerceiveNewEvent()
  SG ← ContextManager.GetSalientSocialGroups()
  if SG ≠ ∅ then
    g ← IdentityManager.SelfCategorisation(SG, self)
    if e.ResponsibleAgent ∈ g then
      e.ResponsibleAgent ← g.Name
      self ← g.Name
    end if
  end if
  AV ← Appraisal.DetermineVariables(e)
  E ← Appraisal.GenerateEmotions(AV, self)
  se ← StrongestEmotion(E)
  for all c ∈ Actuators.GetEmotionChannels() do
    Express(se, c)
  end for
end while
```

Individual-based Emotions

# How?

## Group-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

## Individual-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

# How?

## Group-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

Event(P3,IncreasePoints(Trick,11))

## Individual-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

Event(P3,IncreasePoints(Trick,11))

# How?

## Group-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

Event(P3,IncreasePoints(Trick,11))

{T1,T2} ← ContextManager.GetSalientSocialGroups()
T1 ← IdentityManager.SelfCategorisation(SG, self)

If P3 ∈ T1
Then,
- Event(**T1**,IncreasePoints(Trick,11))
- Self ← **T1**



## Individual-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

Event(P3,IncreasePoints(Trick,11))

# How?

## Group-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

Event(P3,IncreasePoints(Trick,11))

Appraisal

Pride*



## Individual-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

Event(P3,IncreasePoints(Trick,11))

* Using a OCC Theory of Appraisal

# How?



## Group-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

Event(P3,IncreasePoints(Trick,11))

Appraisal

Pride

## Individual-based Emotions

Assuming the robot is P1 and {P1, P3} ∈ T1

Event(P3,IncreasePoints(Trick,11))

Appraisal

Admiration *

* Using a OCC Theory of Appraisal

# What are their Emotional Responses?

Using the verbal utterances!

Group-based Emotions

Individual-based Emotions

Ex: Partner increases the points

—”We are the best!” (Group Pride)

—“I am impressed with your move!” (Admiration)

# What are their Emotional Responses?

Using the verbal utterances!

Group-based Emotions

Individual-based Emotions

Ex: Partner increases the points

—”We are the best!” (Group Pride)

—“I am impressed with your move!” (Admiration)

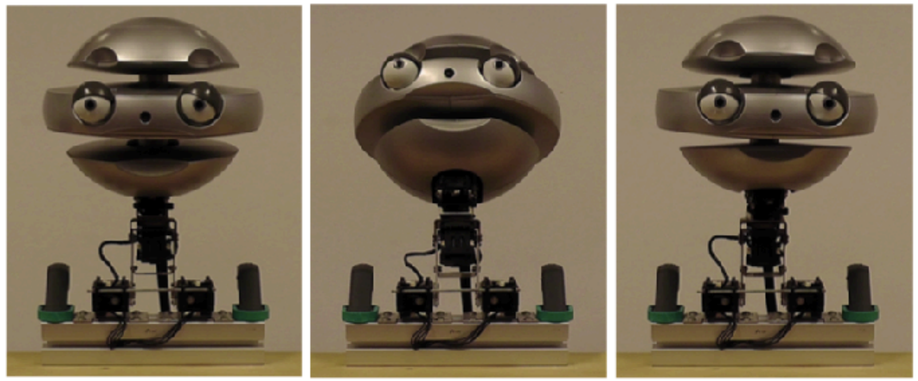
Ex: Robot decreased the points

—”Sorry partner, for this unfortunate move.” (Group Shame)

—“I am so ashamed of my move... ” (Individual Shame)

# What are their Emotional Responses?

Using the physical posture!



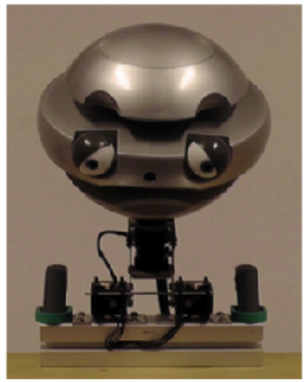
(a) Joy



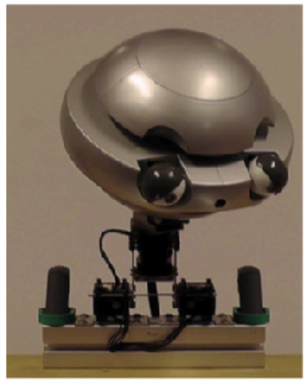
(b) Pride



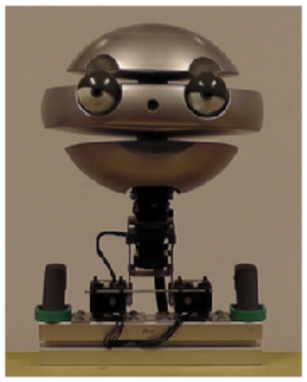
(c) Admiration



(d) Distress



(e) Shame



(f) Reproach

# User Study

# Measures

Towards the robotic partner:

- Group Identification [1] (Satisfaction, Solidarity)
- Godspeed [2] (Anthropomorphism, Animacy, Likeability, Perceived Intelligence)
- Group trust [3]

[1] **Leach**, Colin Wayne, et al. "**Group-level self-definition and self-investment: a hierarchical (multicomponent) model of in-group identification**." *Journal of personality and social psychology* 95.1 (2008): 144.
[2] **Bartneck**, Christoph, et al. "**Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots**." *International journal of social robotics* 1.1 (2009): 71-81.
[3] **Allen**, Kathleen, Richard Bergin, and Kenneth Pickar. "**Exploring trust, group satisfaction, and performance in geographically dispersed and co-located university technology commercialization teams**." *VentureWell. Proceedings of Open, the Annual Conference*. National Collegiate Inventors & Innovators Alliance, 2004.

# Procedure

- Briefing and consent form
- Explain the rules and play an example game (without the robots)
- Random draw to assign the robotic partner
- 3 games with the robots
- Questionnaire
- Random draw of a cinema ticket
- Debriefing

45 minutes

MULTI
TACTION
MultiTaction Cell 55"
protective calibration sheet
GLIN

# Sample

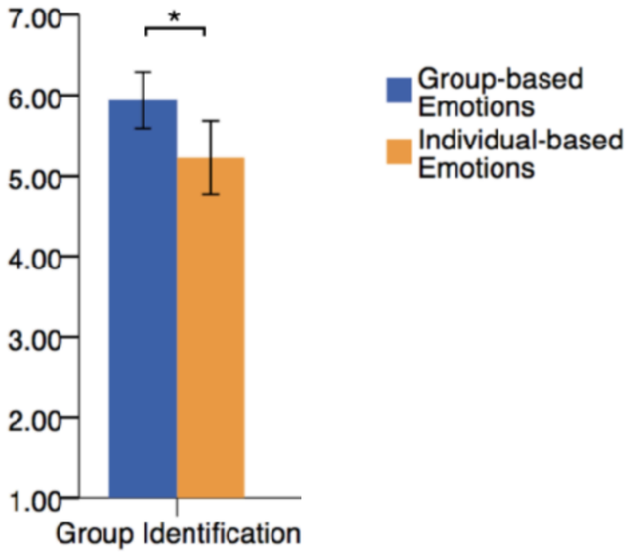
- 48 university students (24 sessions)
  - 33 males and 15 females
  - [19 - 33] years old (M = 25.02 ± 2.98)

# Results

- Game scores were balanced
  - Team GbE won 10 times, lost 11 and tied 3
  - Team IbE won 11 times, lost 10 and tied 3
- The number of expressed emotions between robots was balanced
  - However, there were more positive emotions than negative emotions

# Results - Group Identification

- Participants had **significantly higher** levels (U = 175.5, p = 0.02, r = 0.335) of **Group Identification** towards the robotic partner with GbE than towards the robotic partner with IbE.

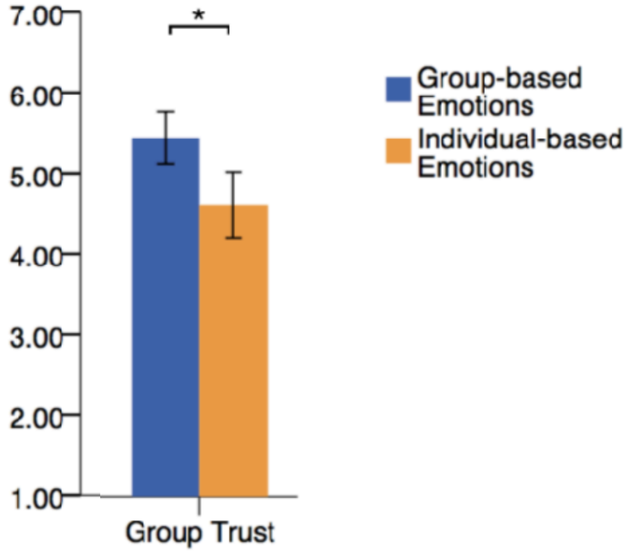


# Results - Group Identification

- There was a **non-significant correlation** ($r_s = 0.153$, $p = 0.30$) between **the number of points** of the team and the level of **Group Identification**

# Results - Group Trust

- Participants had **significantly higher** levels (U = 148, $p < 0.01$, r = 0.417) of **Group Trust** towards the robotic partner with GbE than towards the robotic partner with IbE.



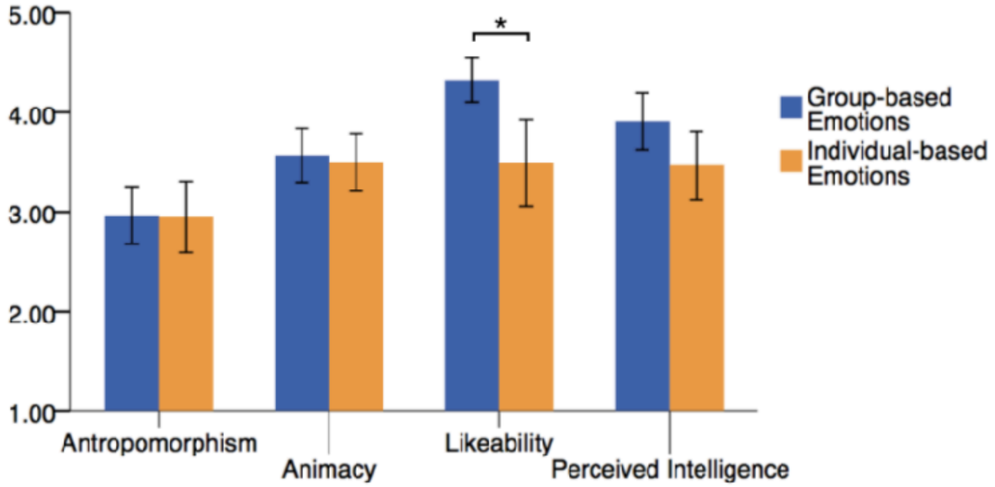
# Results - Group Trust

- There was a **non-significant correlation** ($r_s = 0.158$, $p = 0.28$) between **the number of points** of the team and the level of **Group Trust**.

# Results - Perception of the Robot

- Results showed **no significant diferences** in:
  - Perceived Intelligence (U = 200, p = 0.07),
  - Animacy (U = 275, p = 0.79),
  - Anthropomorphism (U = 276, p = 0.80).

- Participants attributed **significantly higher** levels of **Likeability** to robotic partner with GbE than the robotic partner with IbE.



# Results - Perception of the Robot

- There was a **strong**, **positive**, and statistically **significant correlation** between **Group Identification** and:
  - Anthropomorphism ($r_s = 0.529$, $p < 0.01$)
  - Animacy ($r_s = 0.318$, $p = 0.03$)
  - Likeability ($r_s = 0.606$, $p < 0.01$)
  - Perceived Intelligence ($r_s = 0.595$, $p < 0.01$)

# Discussion

- **H1**: Participants will have a **stronger Group Identification** with a robotic partner that expresses GbE.

# Discussion

**H1**: Participants will have a **stronger Group Identification** with a robotic partner that expresses GbE.

# Discussion

**H1**: Participants will have a **stronger Group Identification** with a robotic partner that expresses GbE.

- **H2**: Participants will have a more **positive perception** of a robotic partner that expresses GbE.

# Discussion

**H1**: Participants will have a **stronger Group Identification** with a robotic partner that expresses GbE.

**H2**: Participants will have a more **positive perception** of a robotic partner that expresses GbE.

# Discussion

**H1**: Participants will have a **stronger Group Identification** with a robotic partner that expresses GbE.

**H2**: Participants will have a more **positive perception** of a robotic partner that expresses GbE.

- **H3**: Participants will have a **higher degree of Group Trust** with a robotic partner that has GbE.

# Discussion

**H1**: Participants will have a **stronger Group Identification** with a robotic partner that expresses GbE.

**H2**: Participants will have a more **positive perception** of a robotic partner that expresses GbE.

**H3**: Participants will have a **higher degree of Group Trust** with a robotic partner that has GbE.

# Conclusions

# Conclusions

- We defined a **model of GbE** for social robotic partners
- **User study with 2 fully autonomous robots** using the model to generate GbE or IbE
- Our findings can improve interactions and create **effective collaborations in HRI**
- **GbE revealed a promising role on the design of social robotic partners**

# Thank you!

## Questions?