

AA 598B



Socially Compliant Mobile Robot Navigation via Inverse Reinforcement Learning

14 Oct 2024

Presenters:
Shubham Mittal
Anubhav Vishwakarma



Robots navigating while respecting social norms?

Ref: Chen, Y. F. et al. (2019)

**How robots can
learn to behave in
a socially
compliant way?**

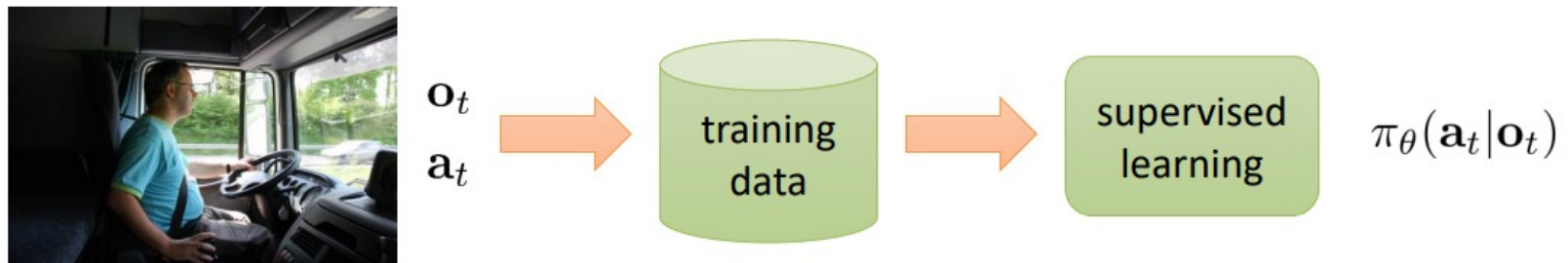
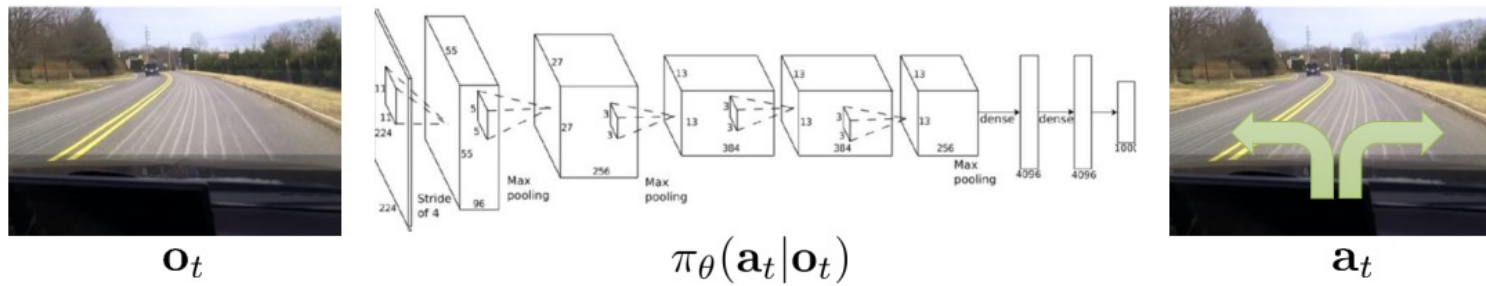


Basic Approach – Learning from Human Behavior

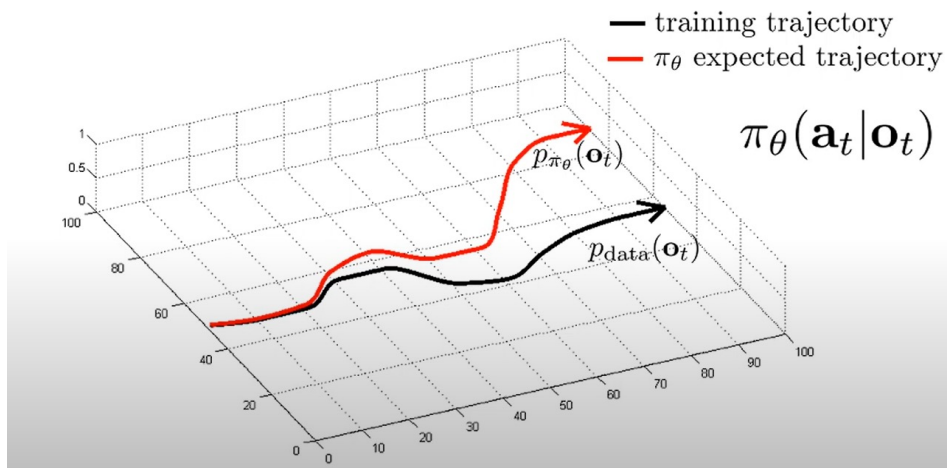
- One potential solution is to observe human behavior and learn from it
- Human navigation involves both **continuous** (e.g., speed adjustments) and **discrete decisions** (e.g., passing left or right of obstacles)
- By learning these behaviors, robots can emulate socially compliant movements.

Early Approaches to Learning from Humans

Behavior Cloning



Ref: Atkeson & Schaal (1997)



**Does it
work?
No!**

**Distributional Shift
Compounding error!**

Transitioning to Inverse Reinforcement Learning (IRL)

- Learning the **underlying goal or intention** behind human actions
- Rather than mimicking actions directly, it infers the reward function from **demonstrations** that humans are optimizing for
- Once the reward function is learned, robots can generate behavior by optimizing this learned reward, adapting to **new situations** rather than just copying trajectories.

IRL Techniques: Feature Expectation Matching

- To learn from observations, we model behavior as a probability distribution
- Define a **feature vector** $f : X \rightarrow \mathbb{R}^n$:

$$f_D = \frac{1}{|D|} \sum_{x_k \in D} f(x_k)$$

- Objective is to find the distribution $p(x)$ that matches these empirical feature values in expectation:

$$E_{p(x)}[f(x)] = f_D$$

Limitations:

- Different policies can result in identical feature counts
- Multiple reward functions can yield the same optimal policy, leading to uncertainty in identifying the true behaviors
- When sub-optimal behaviors are observed, mixtures of policies may be necessary to match feature counts, complicating the learning process without a clear method to resolve this ambiguity.

IRL Techniques: Maximum Entropy IRL

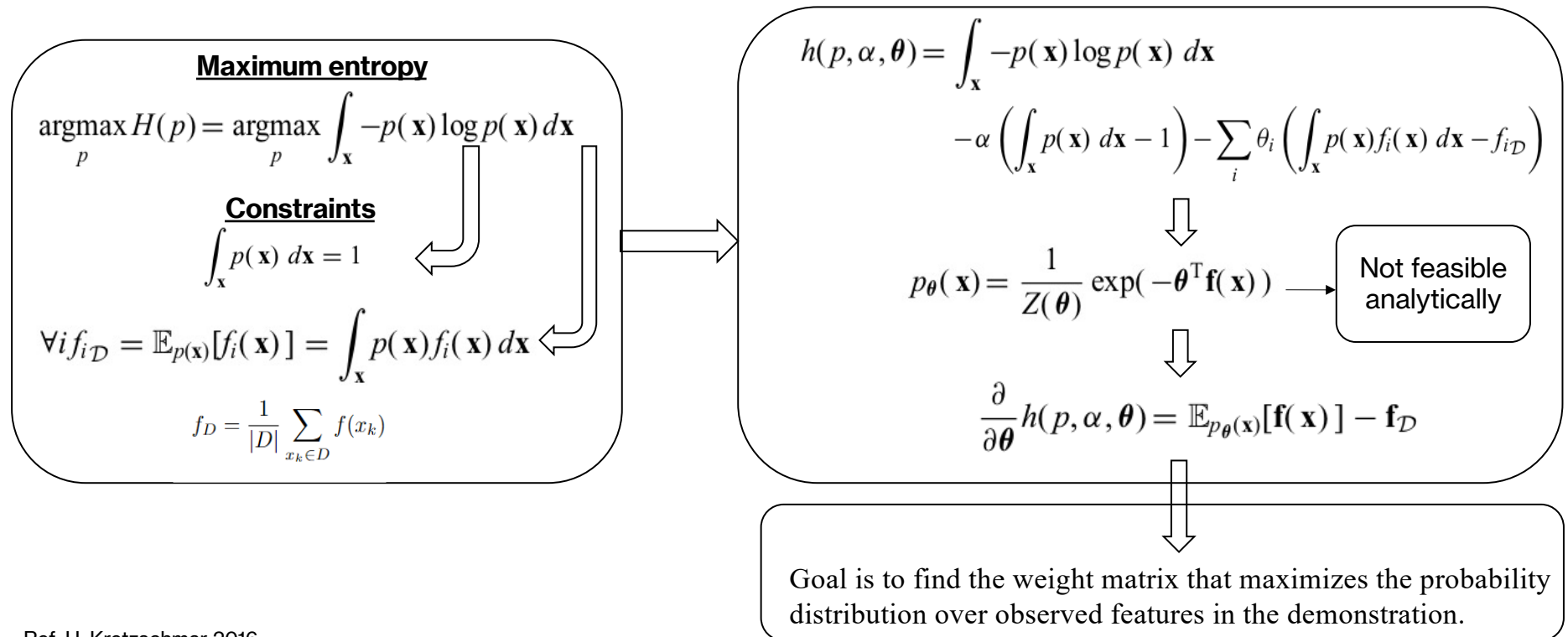
- Apply the principle of maximum entropy to define the best-fitting distribution.

$$\operatorname{argmax}_p H(p) = \operatorname{argmax}_p \int_x -p(x) \log p(x) dx$$

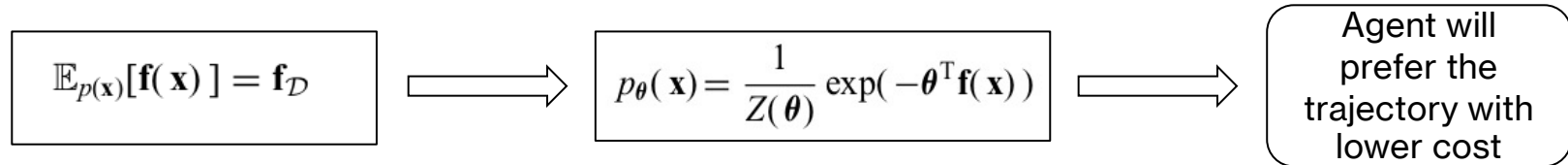
Reason for Using Entropy:

- Ensures that the learned distribution captures the **uncertainty** of human behavior.
- Prevents overfitting to specific demonstrations, allowing for a broader representation of possible actions.

Modelling Pedestrian Navigation Behavior



Continuous Navigation Decisions



Continuous Navigation Features

$$f_{\text{time}}^A(\mathbf{x}) = \sum_{a \in A} \int_t 1 \, dt$$

Time

$$f_{\text{velocity}}^A(\mathbf{x}) = \sum_{a \in A} \int_t \|\dot{x}^a(t)\|^2 \, dt$$

Velocity

$$f_{\text{acceleration}}^A(\mathbf{x}) = \sum_{a \in A} \int_t \|\ddot{x}^a(t)\|^2 \, dt$$

Acceleration

$$f_{\text{obstacle}}^A(\mathbf{x}) = \sum_{a \in A} \int_t \frac{1}{\|x^a(t) - o_{\text{closest}}^a(t)\|^2} \, dt$$

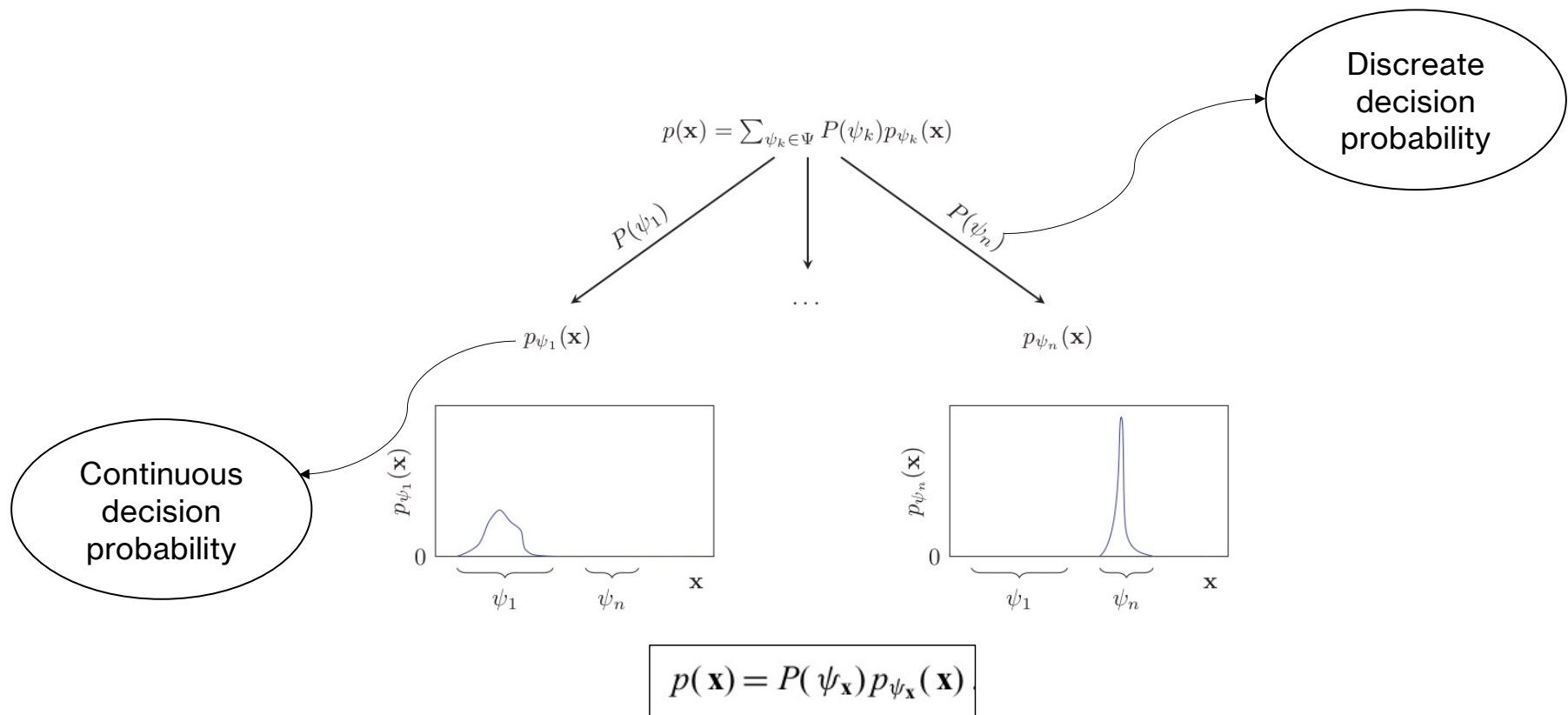
Clearance to other agents

$$f_{\text{distance}}^{\text{phys}}(\mathbf{x}) = \sum_{a \in A} \sum_{b \neq a} \int_t \frac{1}{\|x^a(t) - x^b(t)\|^2} \, dt$$

Collision avoidance w.r.t static obstacle

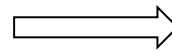
- **Limitation of continuous features:** Highly peaked probability distribution. The regions of smooth, collision-free trajectories are surrounded by regions of very low probability.

Joint mixture distribution

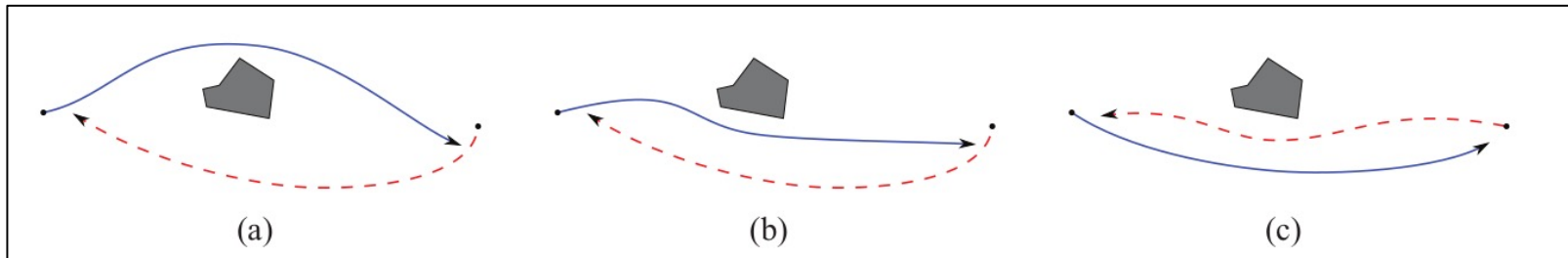
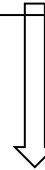


Discrete Navigation Decisions

$$P_{\theta^{\text{hom}}}(\psi) = \frac{1}{Z(\theta^{\text{hom}})} \exp(-\theta^{\text{hom}^T} \mathbf{f}^{\text{hom}}(\psi))$$

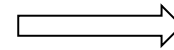


Apply gradient based optimization to get the feature matrix for **homotopy class**



Discrete Navigation Decisions

$$P_{\theta^{\text{hom}}}(\psi) = \frac{1}{Z(\theta^{\text{hom}})} \exp(-\theta^{\text{hom}^T} \mathbf{f}^{\text{hom}}(\psi))$$



Apply gradient based optimization to get the feature matrix for homotopy class

Constraints

$$\mathbb{E}_P[\mathbf{f}^{\text{hom}}] = \mathbf{f}_D^{\text{hom}} = \sum_{\mathbf{x} \in \mathcal{D}} \frac{\mathbf{f}^{\text{hom}}(\psi_{\mathbf{x}})}{|\mathcal{D}|},$$

$$\mathbb{E}_{P(\psi)}[\mathbf{f}^{\text{hom}}] = \sum_{\psi \in \Psi} P(\psi) \mathbf{f}^{\text{hom}}(\psi)$$

Discrete Navigation Features

Passing Left vs right

Group Behavior

Most likely composite trajectory

Follow Up Projects

- **Autonomous Racing with Competitive Strategies**

Objective: Apply the IRL-based trajectory planning model to autonomous racing, where the robot must not only navigate but also compete and strategize against opponents.

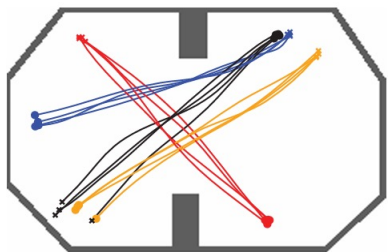
Approach: Use IRL to learn from expert racing drivers, capturing aggressive yet safe maneuvers such as drafting, overtaking, and blocking. The robot can sample competitive trajectories from joint distributions, predicting competitors' moves while optimizing its own path for speed and efficiency.

Challenges: Handling high-speed dynamics, predicting highly unpredictable opponent moves, and achieving precise control in real time.

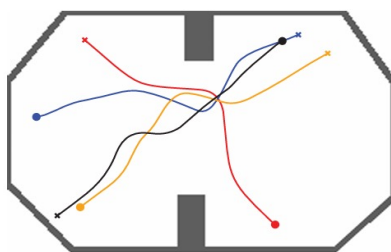
Why this is important at all?



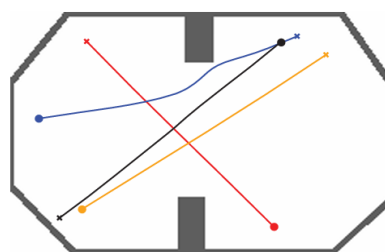
Are they really human like trajectories?



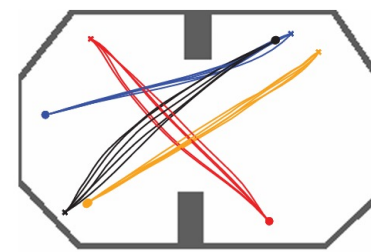
Human Demos



Kuderer et. AI
Maximum Likelihood Joint
Trajectories Corresponding to
the Topological Variants



Social Forces

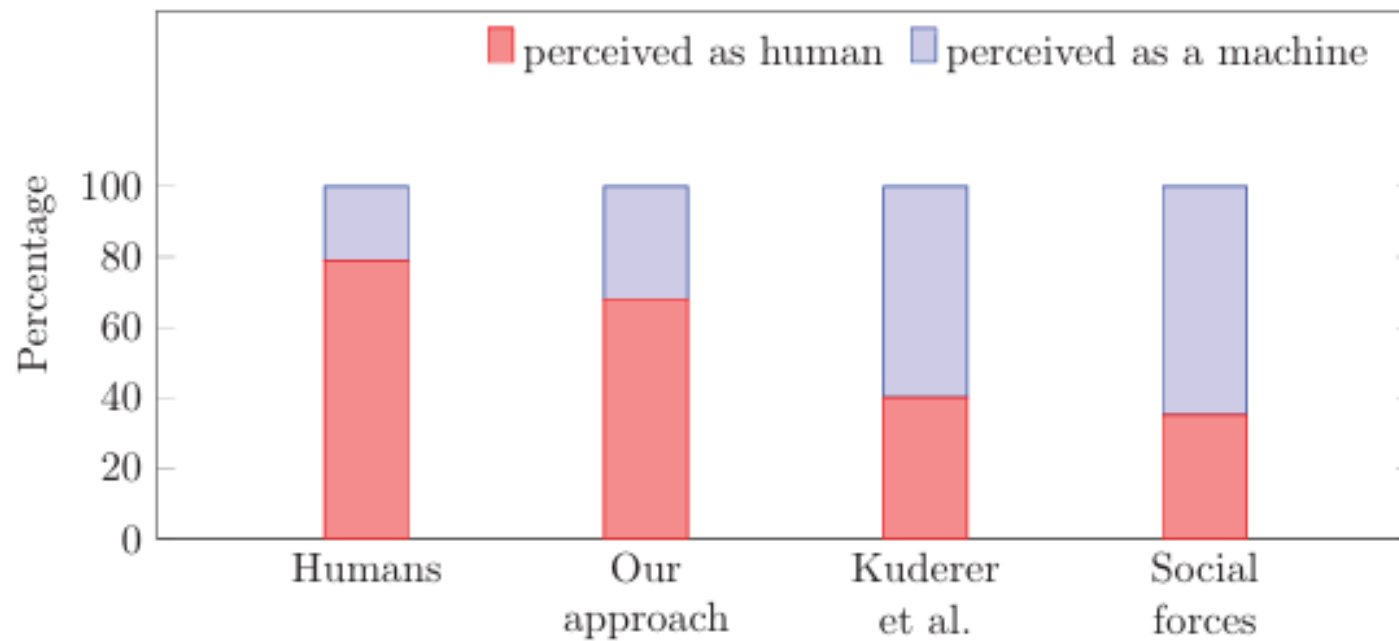


This Paper

Ref: H. Kretzschmar 2016

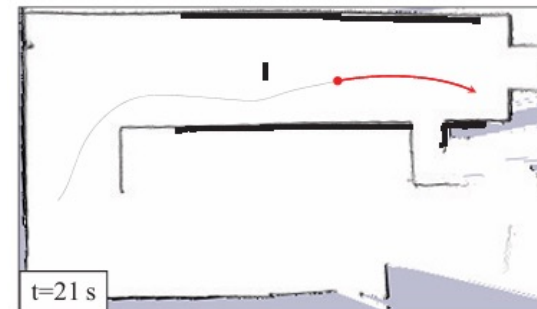
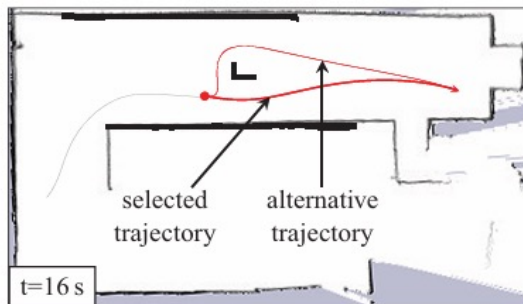
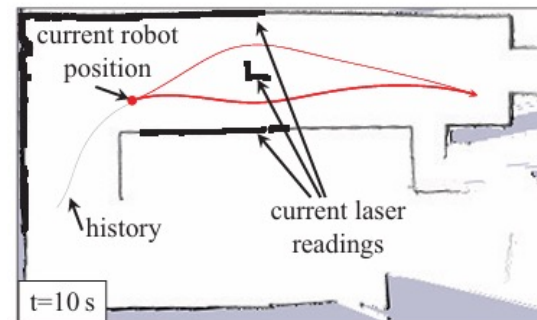
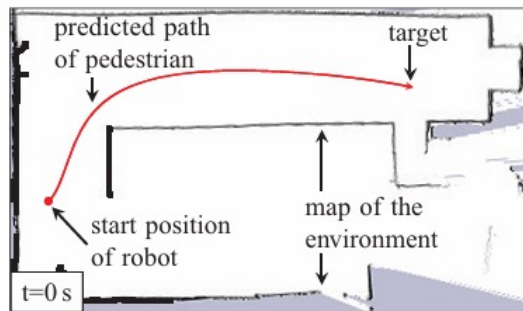
12/16/25

Turing Test



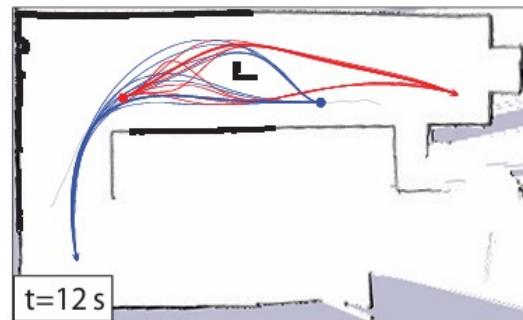
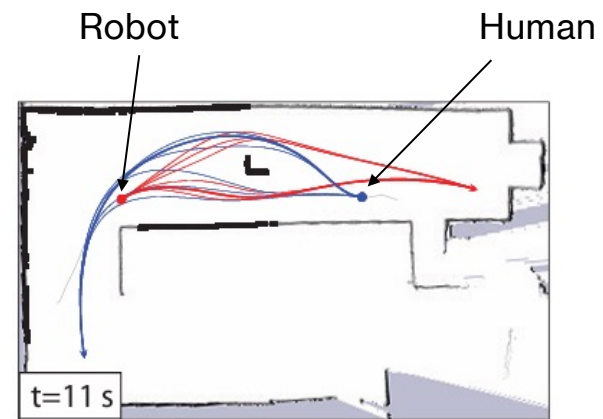
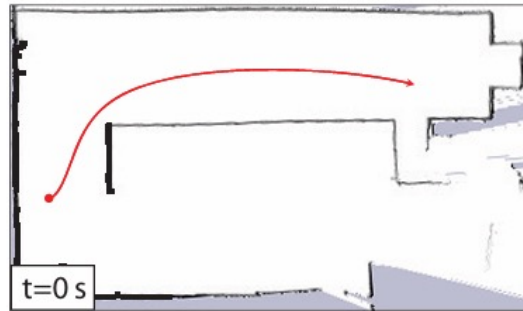
Ref: H. Kretzschmar 2016

Wheel Chair Experiment: Iteration 1



Ref: H. Kretzschmar 2016

Iteration 2:



Ref: H. Kretschmar 2016

Time and Resources Required for Implementation

Time	
Familiarization	1-2 weeks
Data Collection and Preparation	1 weeks
Code-Implementation	6-7 week
Testing and Validation	2-3 weeks
Iteration and Optimization	2-3 weeks
Total Estimate Time	15 weeks

Resources	Cost
Motion Capture System	\$1000
Lidar	\$3000
Mobile Robot	\$3000
Compute	\$4000
Miscellaneous	\$1000
Total Cost	\$12000

Justification for Resources

- **Competitive Advantage:** Enhances our position in the market for social robots.
- **Market Demand:** Aligns with the growing need for safe human-robot interactions.
- **Risk Mitigation:** Proven method reduces potential redesign costs.
- **Collaboration Opportunities:** Opens avenues for partnerships with academic and industry leaders.

Future Impact



Source: MIT CSAIL lecture slide

References

- Chen, Y. F. et al. (2019). Socially aware motion planning with deep reinforcement learning. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2019, 1-7
- Atkeson C and Schaal S (1997) Robot learning from demonstration. In: Proceedings of the fourteenth international conference on machine learning (ICML)
- H. Kretzschmar, M. Spies, C. Sprunk, and W. Burgard, “Socially compliant mobile robot navigation via inverse reinforcement learning,” *The International Journal of Robotics Research*, vol. 35, no. 11, pp. 1289–1307, Sept. 2016
- Kuderer M, Kretzschmar H, Sprunk C, et al. (2012) Featurebased prediction of trajectories for socially compliant navigation. In: Proceedings of robotics: science and systems (RSS).

Thank You :)

Question?

- How would the methods discussed (feature-based vs. IRL) perform in environments with highly unpredictable or chaotic human behavior, such as festivals or large gatherings?
- How scalable do you think these methods are for widespread use? What factors would make them difficult to implement on a large scale (e.g., cost, training data)?

Backup Slides

Strategies for Socially Compliant Robot Navigation

- **Joint Probability Distribution over Composite Trajectories.**
- **Trajectory Sampling, Real-Time Adaptation and Optimization.**
- **Reuse of Optimized Trajectories:** convergence time for optimization algorithms decreases if the starting guess is close to the optimal solution.
- **Scalability of Probability Computations:** The time needed to compute a probability distribution over composite trajectories increases with the travel distance (large environments). Additionally, the accuracy of cooperative behavior predictions diminishes with longer planning horizons.
 - **Short-Term:** The robot uses its learned behavior model to plan for short intervals, generally predicting joint behaviors and adapting to immediate changes in the environment.
 - **Long-Term:** For longer distances or larger environments, the robot employs global path planning algorithms (such as A*). The global planner provides a high-level route based on major waypoints or target areas, which the robot follows and then adjusts locally as needed.

Follow Up Projects

- **Tradeoff between real time navigation performance and learning complex behavior over time.**

Set of Homotopically Distinct Trajectories	Replay buffer in RL
<p>This method is particularly effective in structured or semi-structured environments (e.g., office spaces) where obstacles are relatively static and predictable.</p> <p>Ref: H. Kretschmar 2016</p>	<p>Buffer allows the agent to learn from a variety of situations, which improves generalization in unpredictable environments</p> <p>Ref: Bobak H. Baghi and Gregory Dudek, 2021</p>

Future Impact

- **Robotics in Public Spaces:**

- **Impact:** Greater acceptance of robots in social and public settings.
- **Example:** A future where assistive robots guide people in airports or shopping malls without causing discomfort due to abrupt or socially inappropriate movements.

- **Healthcare and Assistive Robotics**

- **Impact:** Improved quality of care with robots that better understand human social behaviors.
- **Example:** Assistive robots in nursing homes that can anticipate and respond to human movement more intuitively, reducing accidents or discomfort.

- **Human-Robot Collaboration in Industrial Settings**

- **Impact:** Increased efficiency and safety in human-robot collaborative workplaces.
- **Example:** Robots working in factories could navigate around human workers more effectively, reducing accidents and improving team collaboration.

Source: IJRR, and MIT Technology review