

Activity Forecasting

Kris Kitani, Brian Ziebart*, Drew Bagnell, Martial Hebert

Carnegie Mellon University, *The University of Illinois at Chicago

Activity
Recognition

Activity
Forecasting



Activity
Recognition

Activity
Forecasting



Past observed actions

Activity
Recognition

Activity
Forecasting



Past observed actions



Future unobserved actions

Activity Forecasting

Novel scene



[Image from Oh2011]

Activity Forecasting



[Image from Oh2011]

Activity Forecasting



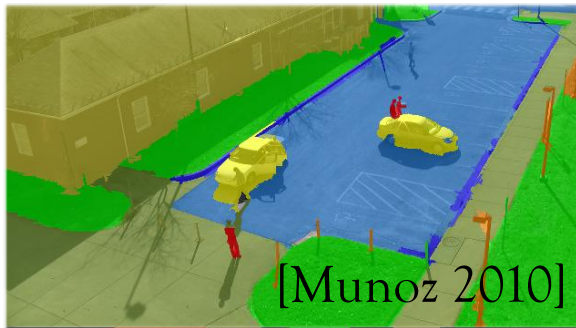
Discovery of subtle yet natural interactions



Demonstrated activity



Physical features

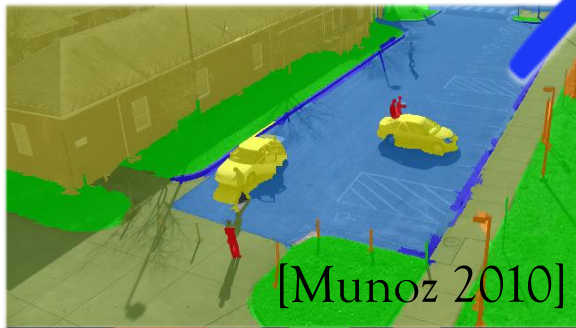


Training data

Demonstrated activity



Physical features



Training data



Person's
decision process

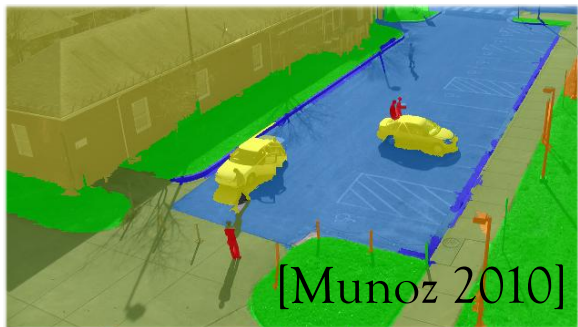
(not scene-specific)



Demonstrated activity



Physical features



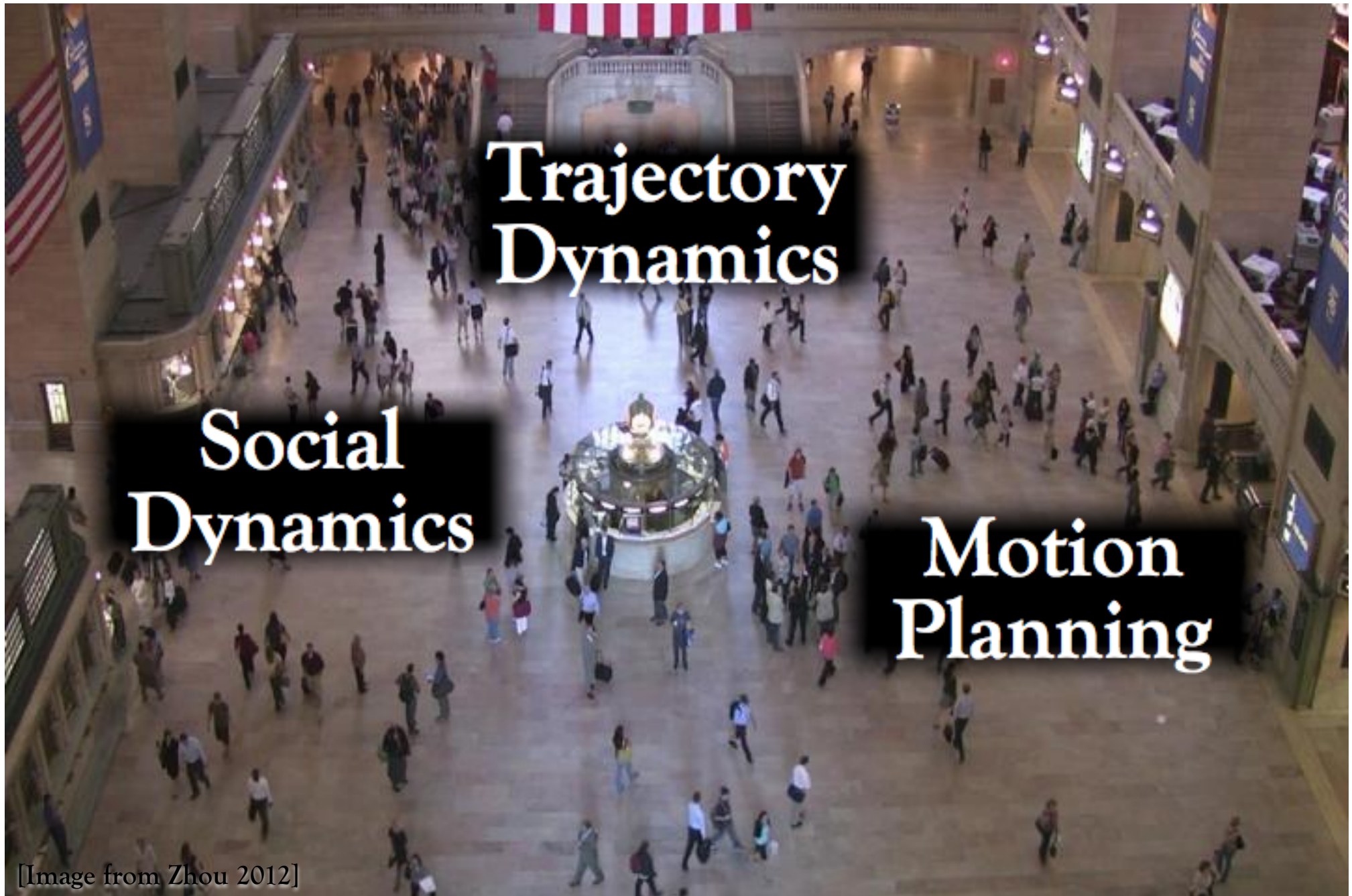
Training data

Person's
decision process
(not scene-specific)



Forecast activities in novel scene

Trajectory-based Activity Analysis



[Image from Zhou 2012]

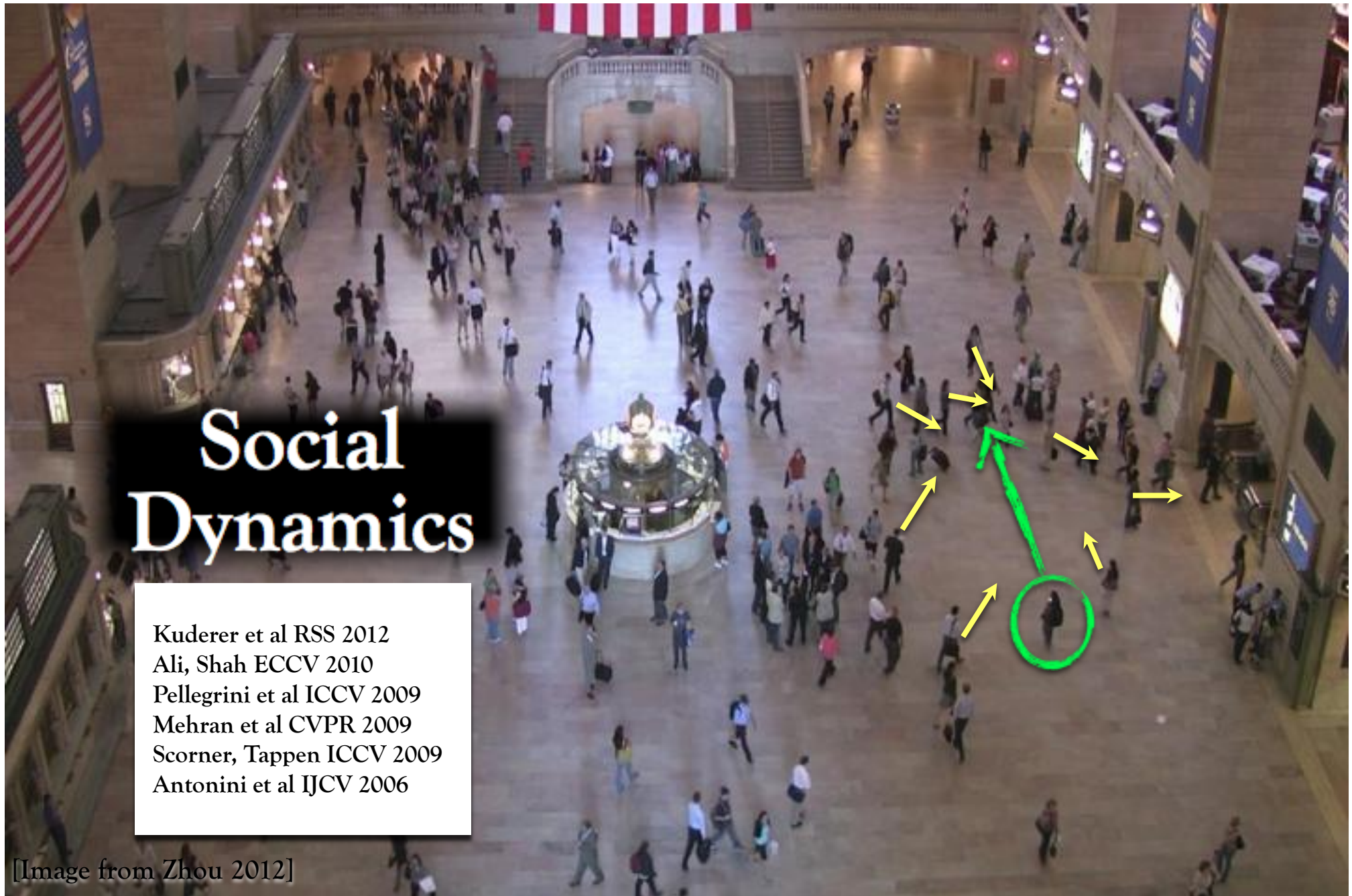
Trajectory-based Activity Analysis

Trajectory Dynamics

Often assume
persistent surveillance

Zhou et al CVPR 2012
Zen, Ricci CVPR 2011
Ali, Shah ECCV 2010
Wang et al CVPR 2008
Morris et al ITS 2008
Hu et al PAMI 2006
Porikli ICME 2004
Buzan et al ICPR 2004
Makris et al CVPR 2004
Hongeng, Nevatia CVPR 2003
Brand, Kettner PAMI 2000
Stauffer et al PAMI 2000
Oliver et al PAMI 2000
Johnson, Hogg BMVC 1995

Trajectory-based Activity Analysis



Social Dynamics

Kuderer et al RSS 2012
Ali, Shah ECCV 2010
Pellegrini et al ICCV 2009
Mehran et al CVPR 2009
Scorner, Tappen ICCV 2009
Antonini et al IJCV 2006

[Image from Zhou 2012]

Trajectory-based Activity Analysis

Decision-theoretic model

- richer model of activity
- not scene-specific

goal

optimal path

obstacle

Motion Planning

Kuderer et al RSS 2012
Gong et al ICCV 2011
Ziebart et al IROS 2010

An aerial, black and white photograph of a large, busy train station. The station is filled with people walking in various directions. In the center, there is a large, ornate fountain. The architecture features high ceilings, multiple levels, and large windows. The overall scene is one of a bustling, public space.

**Trajectory
Dynamics**

**Social
Dynamics**

**Motion
Planning**



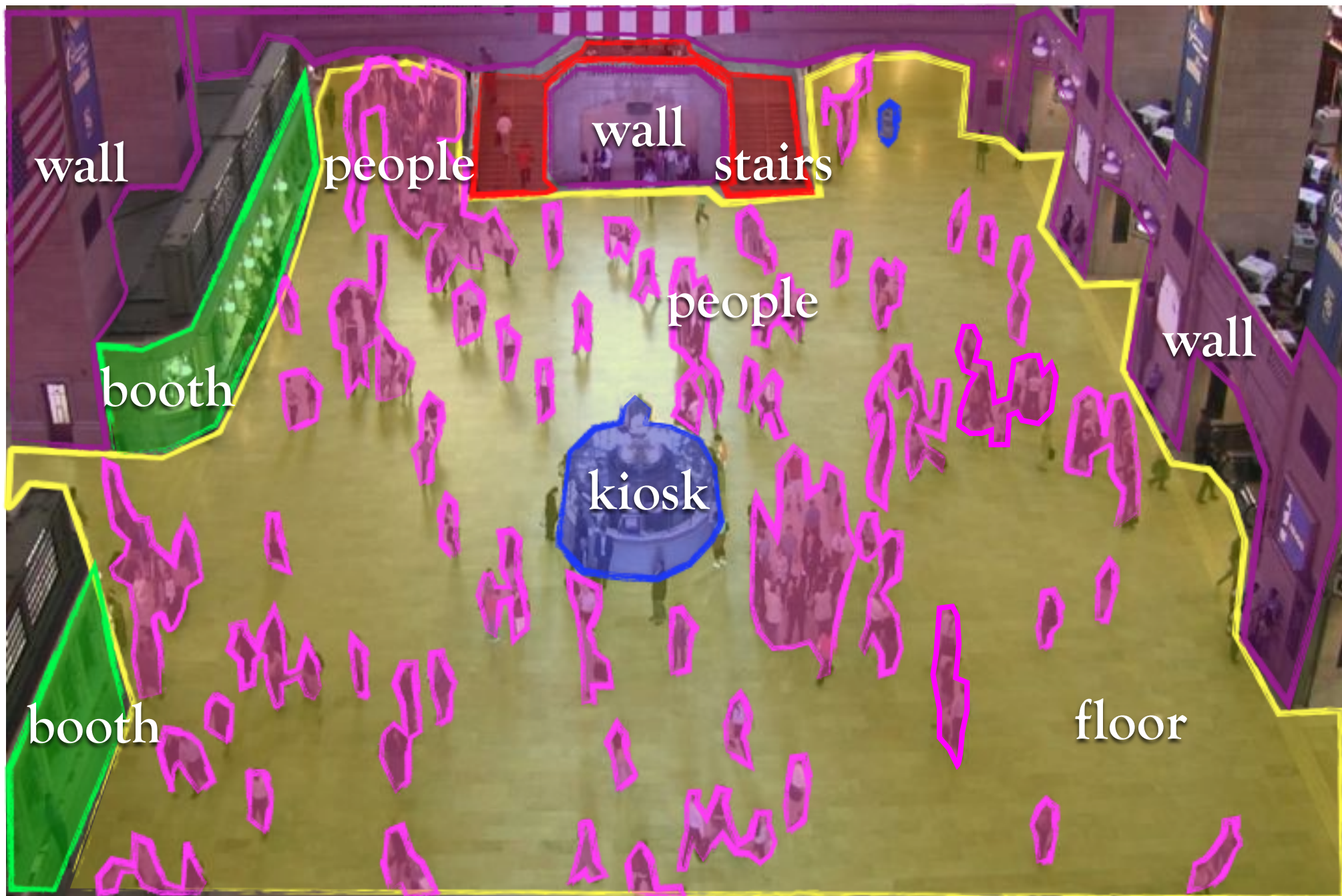
**Trajectory
Dynamics**

**Reduced representations of the
physical world**

**Motion
Planning**



[Image from Zhou 2012]



wall

people

wall

stairs

booth

people

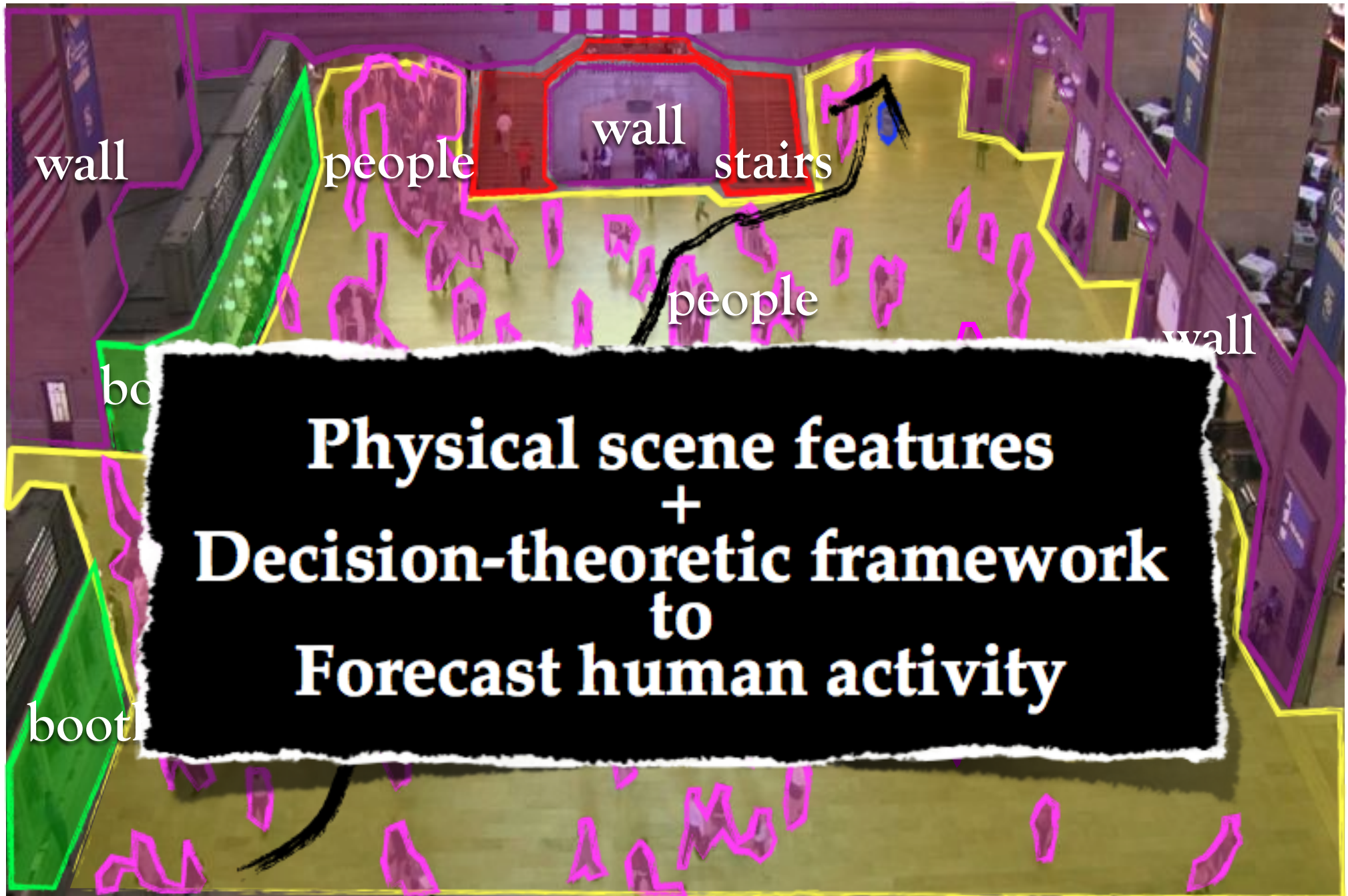
wall

kiosk

booth

floor

Propose to model human activity...



wall

people

wall

stairs

people

wall

bo

**Physical scene features
+
Decision-theoretic framework
to
Forecast human activity**

booth

Towards a decision-theoretic approach



Dynamics

$$p(x' | x)$$

‘motion model’

[Kalman 1960]

[Baum 1966]

Towards a decision-theoretic approach

how?



Dynamics

$$p(x'|x)$$

‘motion model’

[Kalman 1960]

[Baum 1966]

Towards a decision-theoretic approach

how?



Dynamics

$$p(x'|x)$$

‘motion model’

[Kalman 1960]

[Baum 1966]



why?

Decisions

$$p(a|x)$$

‘policy’

[Bellman 1957]

[Howard 1960]

Our Approach:

Activity sequence generated by an Markov Decision Process (MDP)

$$S = \{x_0, a_0, R(x_0), x_1, a_1, R(x_1), \dots, x_g, a_g, R(x_g)\}$$

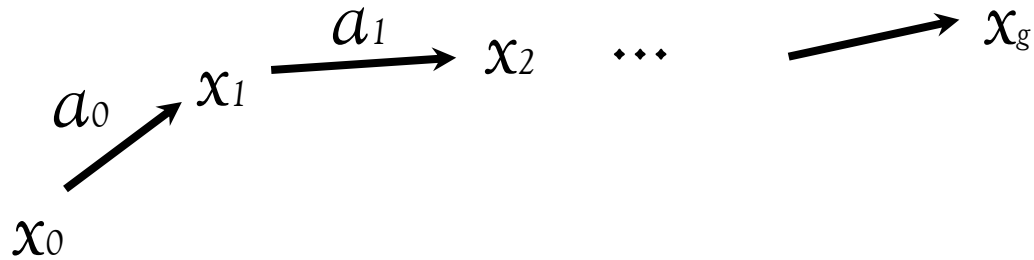
(state, action, reward)

Our Approach:

Activity sequence generated by an Markov Decision Process (MDP)

$$S = \{x_0, a_0, R(x_0), x_1, a_1, R(x_1), \dots, x_g, a_g, R(x_g)\}$$

(state, action, reward)



Our Approach:

Activity sequence generated by an Markov Decision Process (MDP)

$$S = \{x_0, a_0, R(x_0), x_1, a_1, R(x_1), \dots, x_g, a_g, R(x_g)\}$$

(state, action, reward)

Sequence determined by Policy

$$p(a|x)$$

Our Approach:

Activity sequence generated by an Markov Decision Process (MDP)

$$S = \{x_0, a_0, R(x_0), x_1, a_1, R(x_1), \dots, x_g, a_g, R(x_g)\}$$

(state, action, reward)

Sequence determined by Policy

$$p(a|x) \propto \exp \left\{ \underbrace{R(x)}_{\text{Reward}} + \sum_{x'} \underbrace{p(x'|x, a)V(x')}_{\text{Expected future payoff}} \right\}$$

[Ziebart et al 2008]

Our Approach:

Activity sequence generated by an Markov Decision Process (MDP)

$$S = \{x_0, a_0, R(x_0), x_1, a_1, R(x_1), \dots, x_g, a_g, R(x_g)\}$$

(state, action, reward)

Sequence determined by Policy

$$p(a|x) \propto \exp \left\{ \underbrace{R(x)}_{\text{Reward}} + \sum_{x'} p(x'|x, a) \underbrace{V(x')}_{\text{Expected future payoff}} \right\}$$

Policy determined by Reward function

[Ziebart et al 2008]

$$V(x) = \text{soft max}_a Q(x, a)$$

$$Q(x, a) = R(x) + \sum_{x'} p(x'|x, a) V(x')$$

Our Approach:

Activity sequence generated by an Markov Decision Process (MDP)

$$S = \{x_0, a_0, R(x_0), x_1, a_1, R(x_1), \dots, x_g, a_g, R(x_g)\}$$

(state, action, reward)

Sequence determined by Policy

$$p(a|x) \propto \exp \left\{ \underbrace{R(x)}_{\text{Reward}} + \sum_{x'} p(x'|x, a) \underbrace{V(x')}_{\text{Expected future payoff}} \right\}$$

Policy determined by Reward function

[Ziebart et al 2008]

$$V(x) = \text{soft max}_a Q(x, a)$$

$$Q(x, a) = R(x) + \sum_{x'} p(x'|x, a) V(x')$$

Our Approach:

Activity sequence generated by an Markov Decision Process (MDP)

$$S = \{x_0, a_0, R(x_0), x_1, a_1, R(x_1), \dots, x_g, a_g, R(x_g)\}$$

(state, action, reward)

Sequence determined by Policy

$$p(a|x) \propto \exp \left\{ \underbrace{R(x)}_{\text{Reward}} + \sum_{x'} p(x'|x, a) \underbrace{V(x')}_{\text{Expected future payoff}} \right\}$$

Policy determined by Reward function

[Ziebart et al 2008]

$$V(x) = \text{soft max}_a Q(x, a)$$

$$Q(x, a) = \underbrace{R(x)}_{\text{Reward}} + \sum_{x'} p(x'|x, a) V(x')$$

Infer the reward function from observed sequences

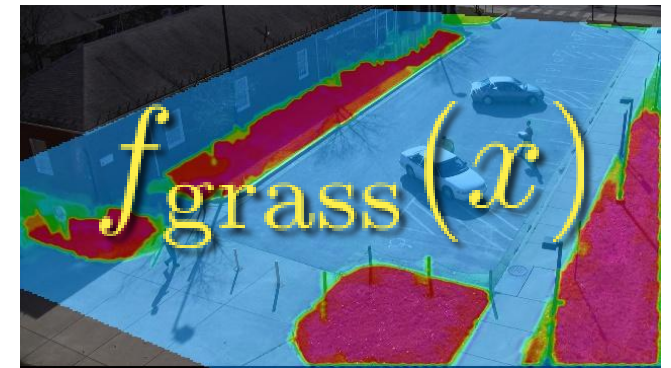
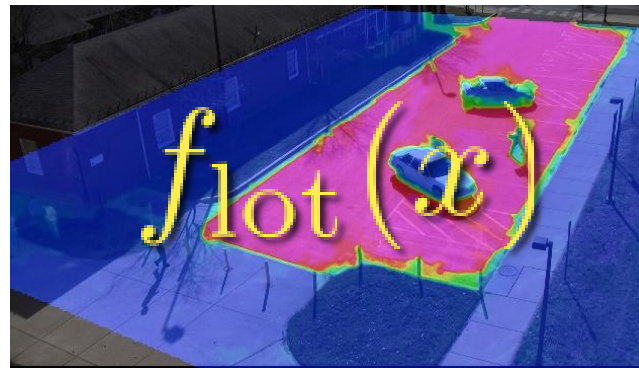
Reward function parameterization

$$R(x; \boldsymbol{\theta}) = \sum_n \theta_n f_n(x)$$

Reward function parameterization

$$R(x; \theta) = \sum_n \theta_n f_n(x)$$

$$R(x; \theta) = \theta_{\text{car}} \cdot f_{\text{car}}(x) + \theta_{\text{lot}} \cdot f_{\text{lot}}(x) + \theta_{\text{grass}} \cdot f_{\text{grass}}(x)$$

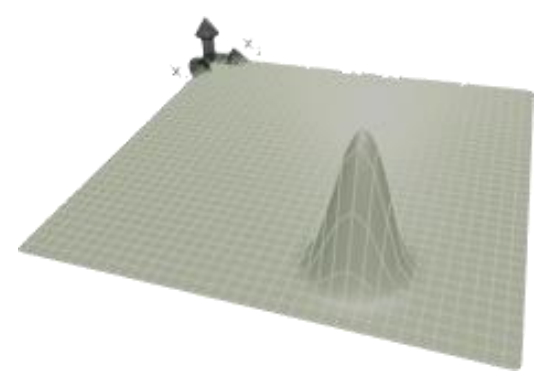
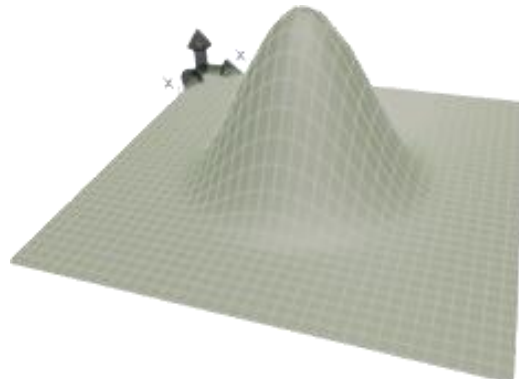
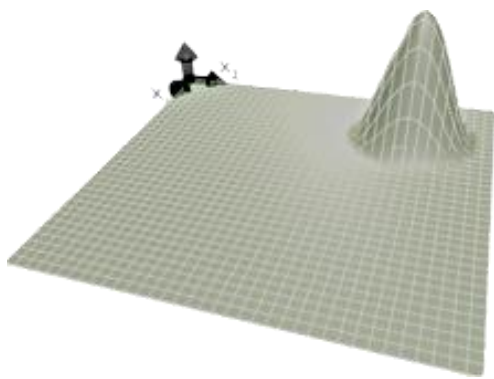


[Munoz et al 2010]

Reward function parameterization

$$R(x; \boldsymbol{\theta}) = \sum_n \theta_n f_n(x)$$

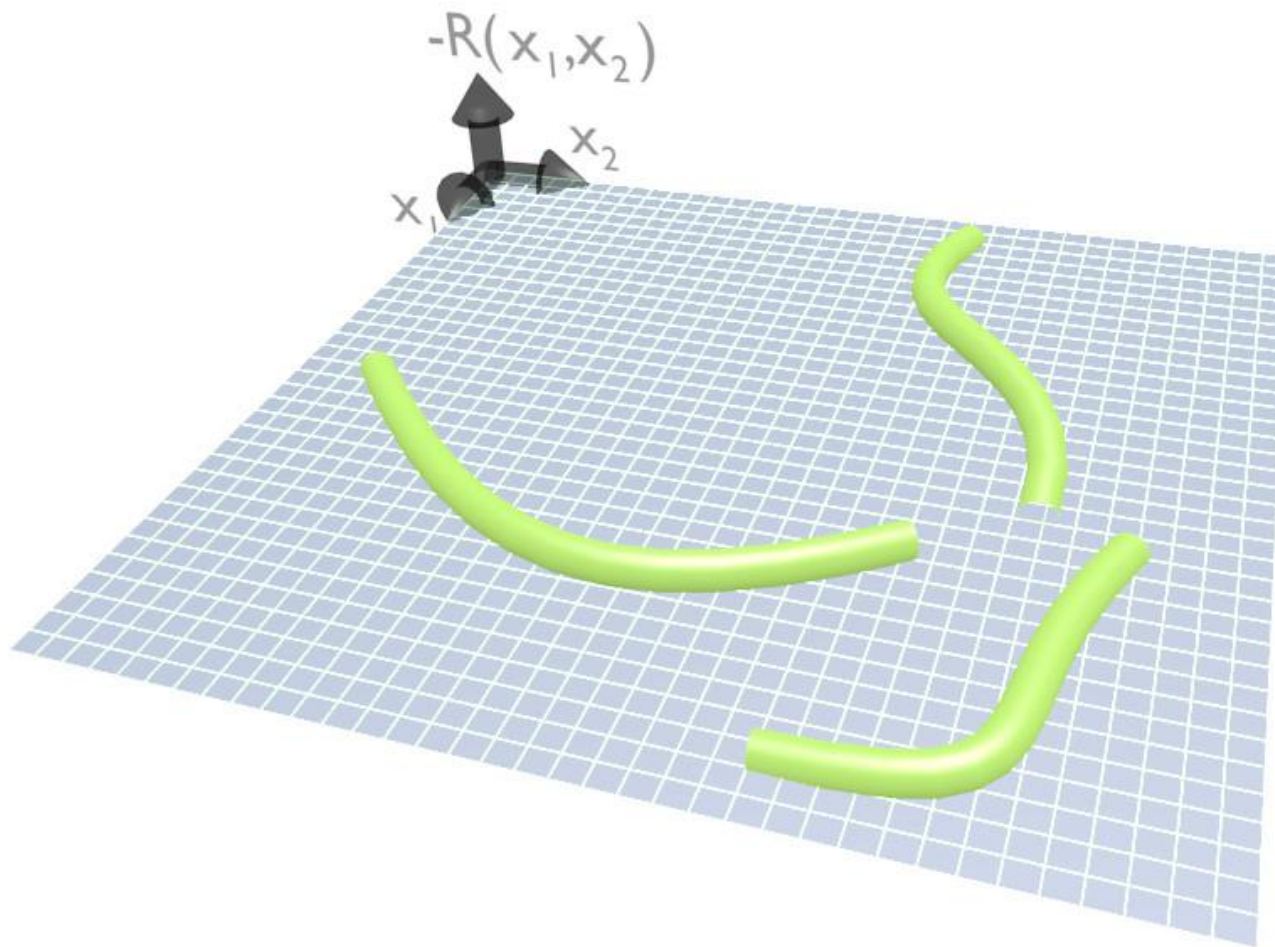
$$R(x; \boldsymbol{\theta}) = \theta_{\text{car}} \cdot f_{\text{car}}(x) + \theta_{\text{lot}} \cdot f_{\text{lot}}(x) + \theta_{\text{grass}} \cdot f_{\text{grass}}(x)$$



Learn the weights of the physical features

Input: Trajectories & feature responses

Output: Reward weights



$$R(x; \theta) =$$
$$\theta_{\text{car}} \cdot f_{\text{car}}(x)$$
$$+$$
$$\theta_{\text{lot}} \cdot f_{\text{lot}}(x)$$
$$+$$
$$\theta_{\text{grass}} \cdot f_{\text{grass}}(x)$$

Three 3D plots showing feature functions for different environments. Each plot has a small car icon on a grid. The first plot shows a single peak labeled $f_{\text{car}}(x)$. The second plot shows a single peak labeled $f_{\text{lot}}(x)$. The third plot shows a single peak labeled $f_{\text{grass}}(x)$. The plots are stacked vertically with plus signs between them.

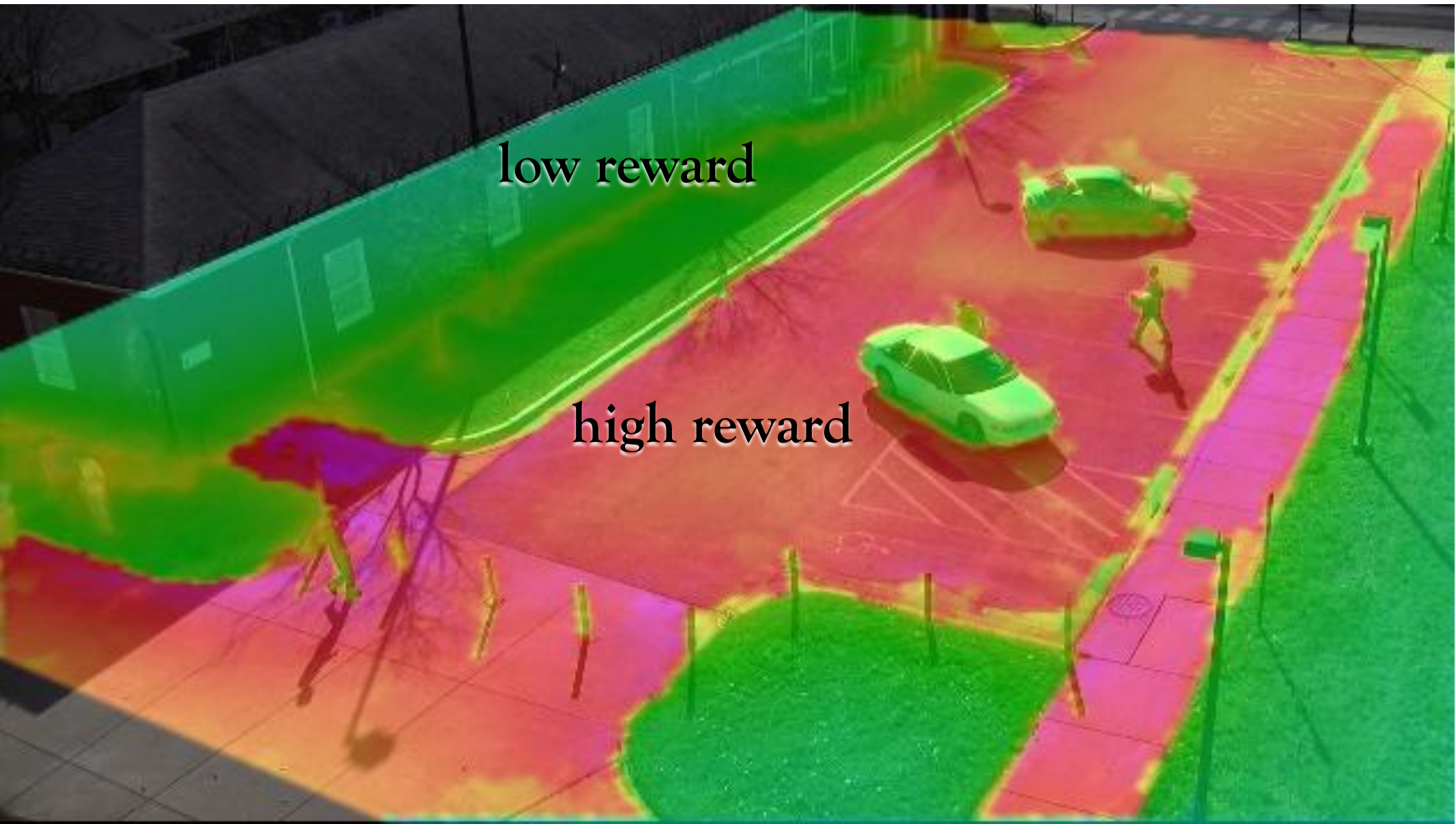
Learn the reward function via

Inverse Optimal Control

[Abbeel & Ng 2004, Ziebart et al 2008]

[Graphics by Paul Vernaza]

Reward Function

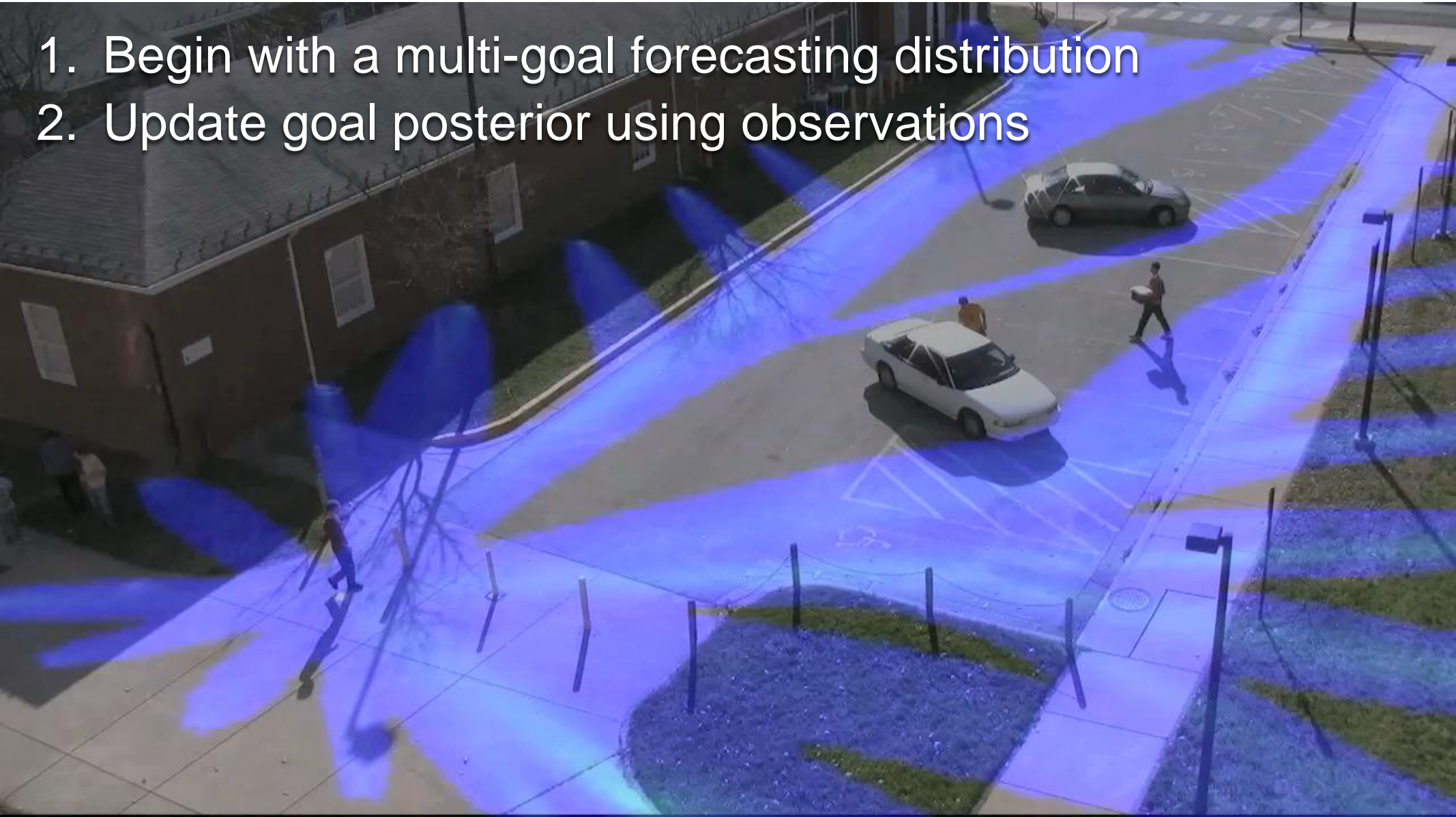


Activity Forecasting



Destination Forecasting

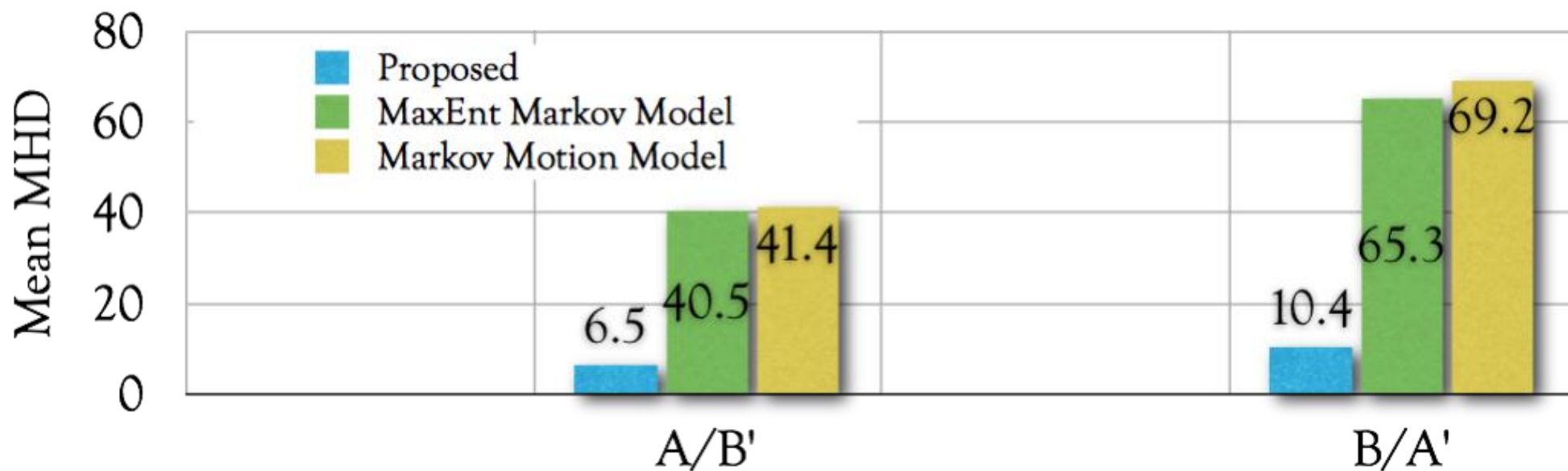
1. Begin with a multi-goal forecasting distribution
2. Update goal posterior using observations



Dataset: 92 videos (A:56 / B:36)
Setup: 80% test, 20% train (3-fold cross validation)
Baselines: Maximum Entropy Markov Model [McCallum'00]
Markov Motion Model [Porikli' 04]
Metrics: Negative log loss,
Modified Hausdorff distance



Forecasting performance



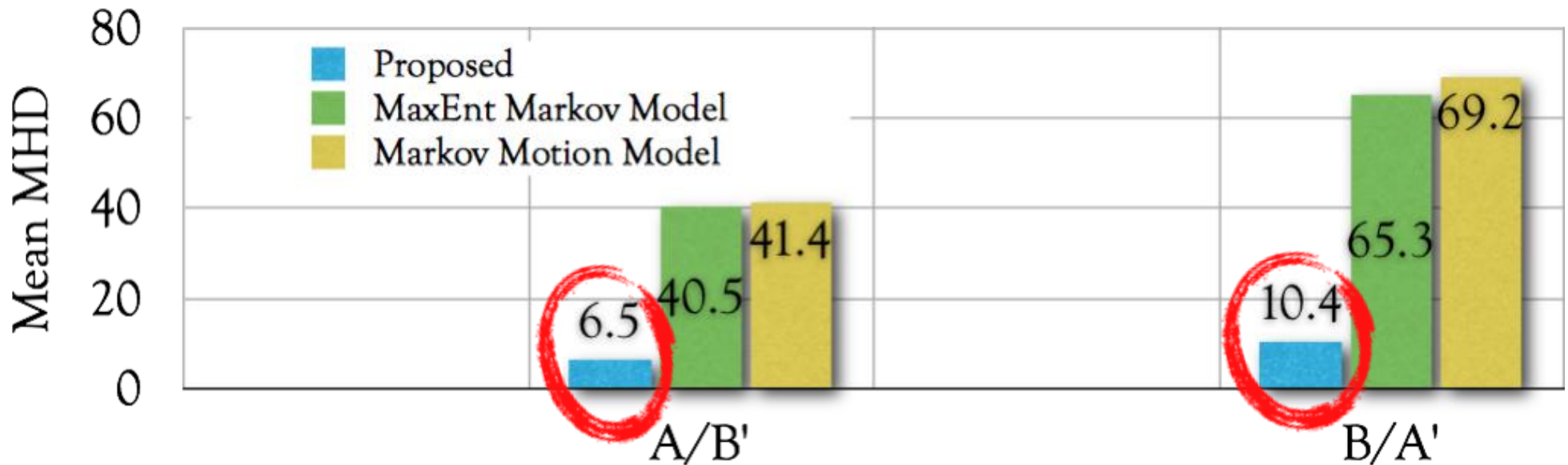
Modified Hausdorff Distance (MHD) [pixels]:

Euclidean distance between observed trajectory and sampled trajectories allowing for small temporal misalignment

Dataset: 92 videos (A:56 / B:36)
Setup: 80% test, 20% train (3-fold cross validation)
Baselines: Maximum Entropy Markov Model [McCallum'00]
Markov Motion Model [Porikli' 04]
Metrics: Negative log loss,
Modified Hausdorff distance



Forecasting performance



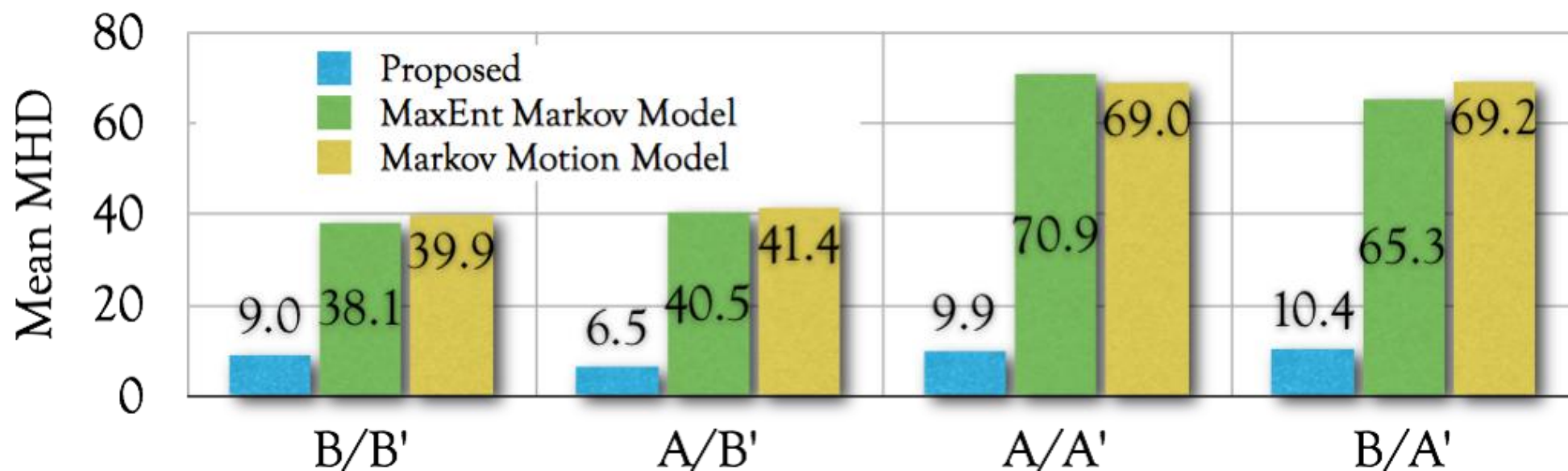
Modified Hausdorff Distance (MHD) [pixels]:

Euclidean distance between observed trajectory and sampled trajectories allowing for small temporal misalignment

Dataset: 92 videos (A:56 / B:36)
Setup: 80% test, 20% train (3-fold cross validation)
Baselines: Maximum Entropy Markov Model [McCallum'00]
Markov Motion Model [Porikli' 04]
Metrics: Negative log loss,
Modified Hausdorff distance



Forecasting performance



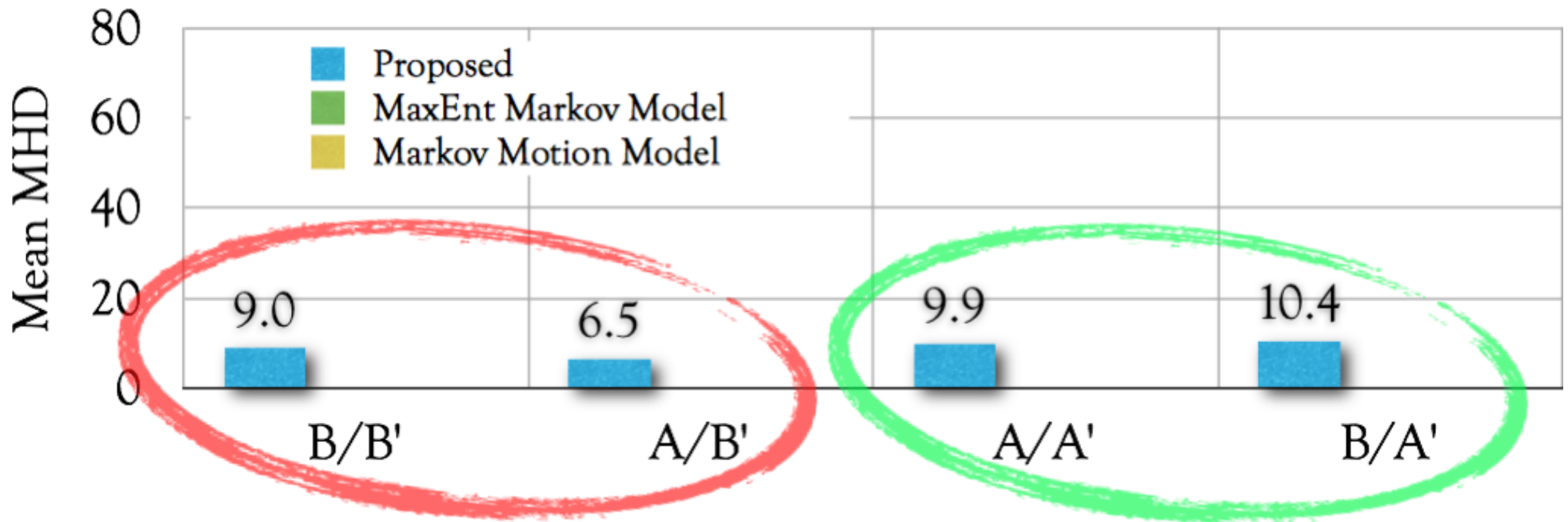
Modified Hausdorff Distance (MHD) [pixels]:

Euclidean distance between observed trajectory and sampled trajectories allowing for small temporal misalignment

Dataset: 92 videos (A:56 / B:36)
Setup: 80% test, 20% train (3-fold cross validation)
Baselines: Maximum Entropy Markov Model [McCallum'00]
Markov Motion Model [Porikli' 04]
Metrics: Negative log loss,
Modified Hausdorff distance



Forecasting performance



Modified Hausdorff Distance (MHD) [pixels]:

Euclidean distance between observed trajectory and sampled trajectories allowing for small temporal misalignment

Forecasting in new scenes



Activity Forecasting

Kris Kitani, Brian Ziebart*, Drew Bagnell, Martial Hebert

Carnegie Mellon University, *The University of Illinois at Chicago