Wifi Localization with Gaussian Processes

Brian Ferris
University of Washington

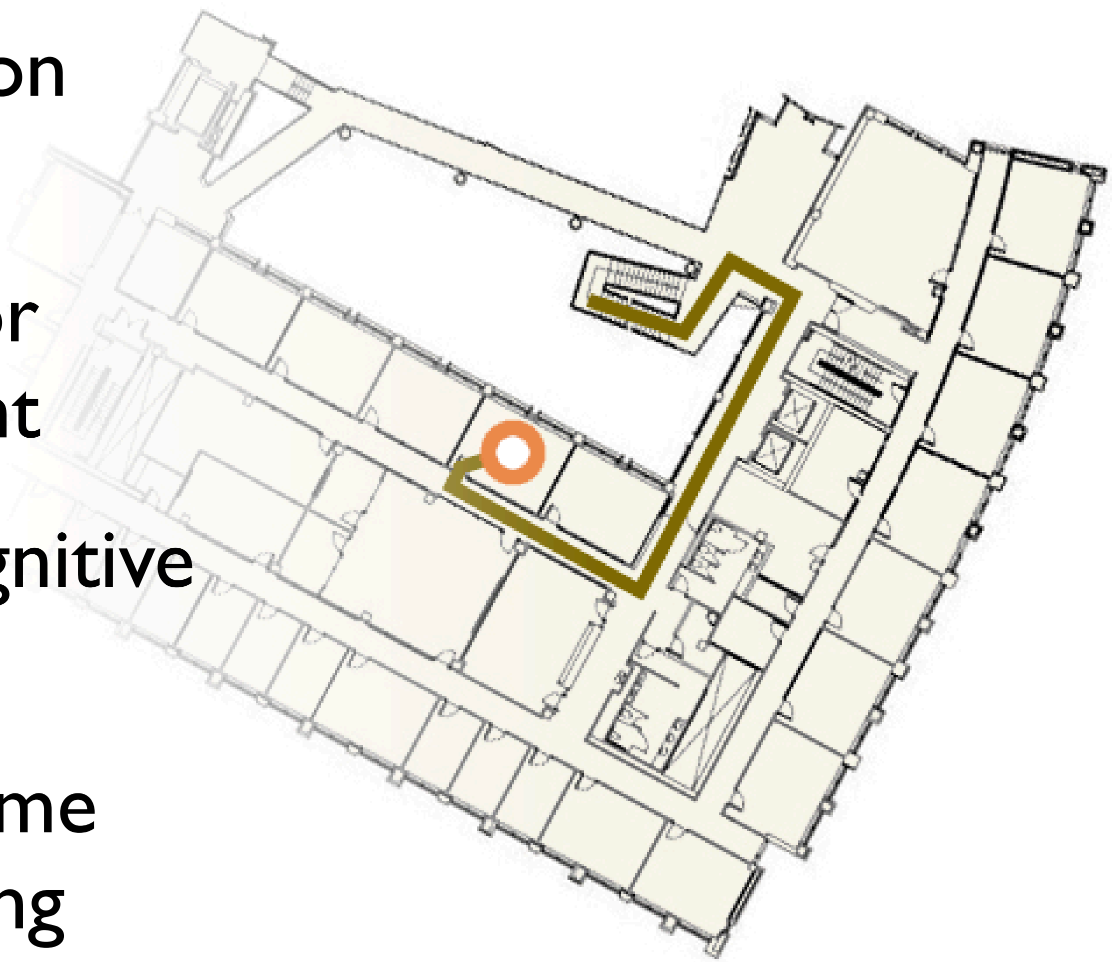
Joint work with:

Dieter Fox | Julie Letchner | Dirk Haehnel

University of Washington Computer Science & Engineering

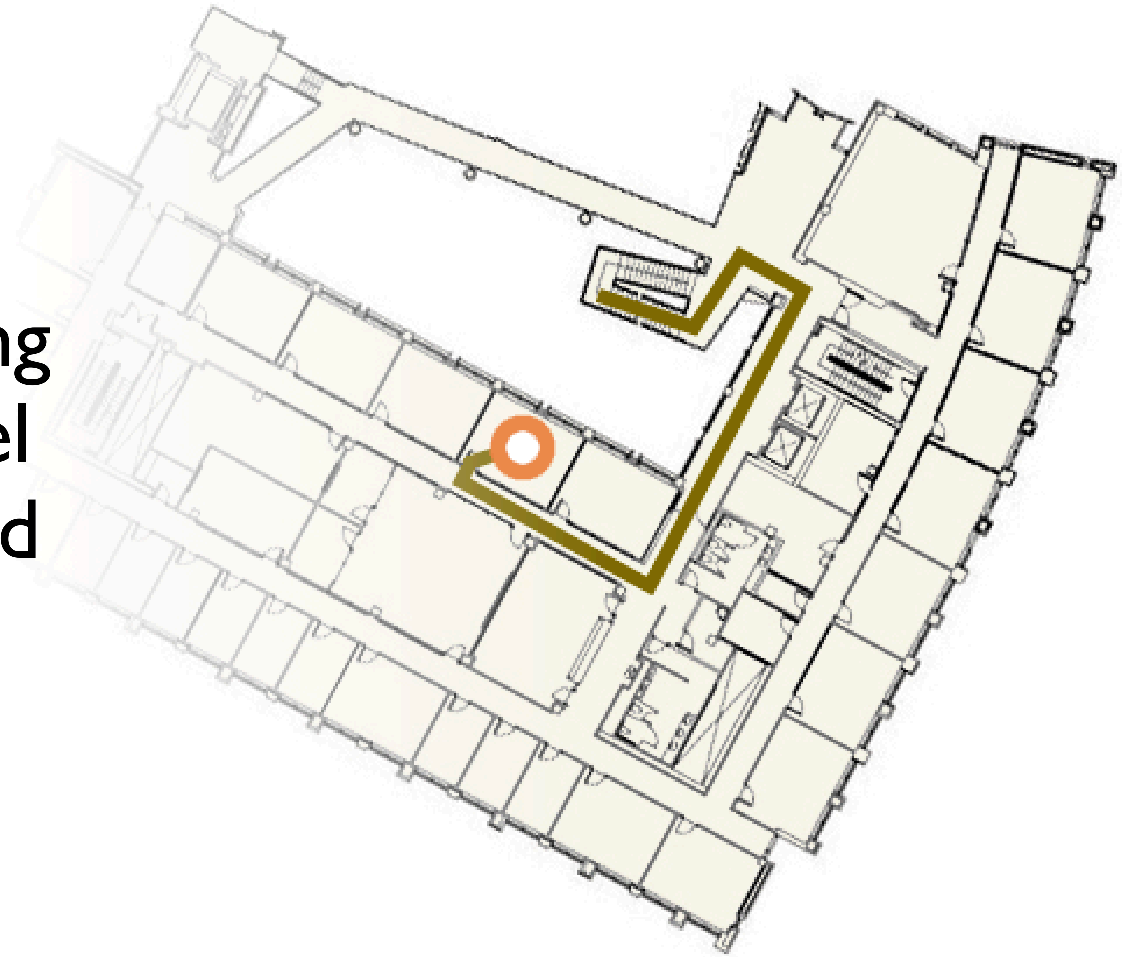
Why Location?

- Assisted Cognition Project:
  - Indoor/outdoor navigation agent
  - Users with cognitive impairments
  - Requires realtime location tracking



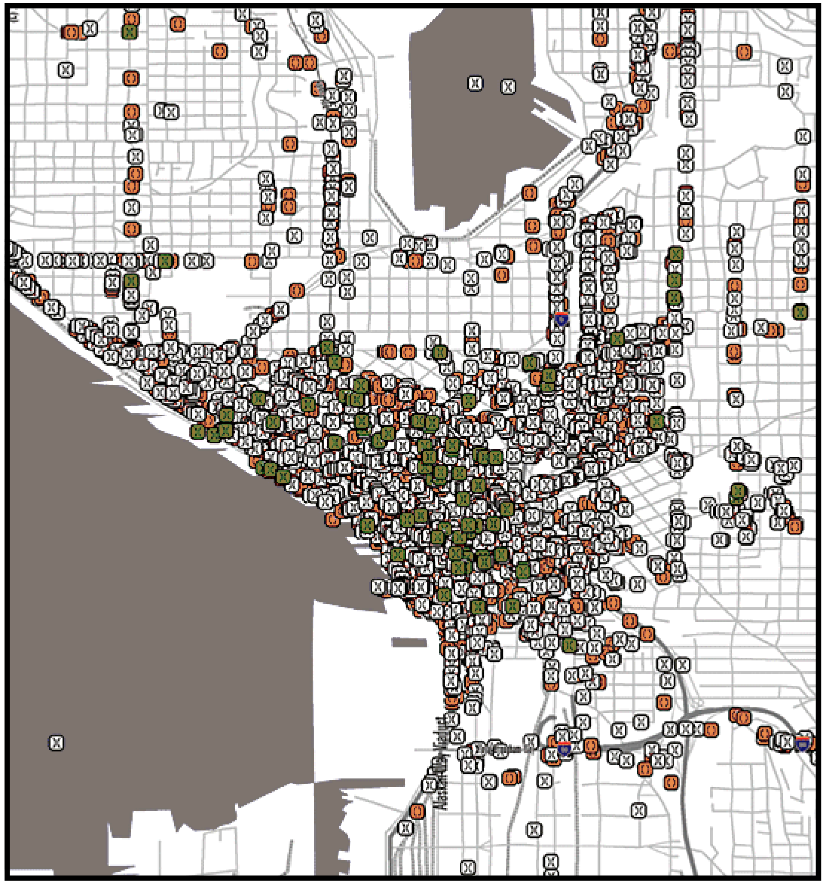
Why Location?

Location is a fundamental building block in higher level state estimation and activity recognition applications



Why Wifi?

- Cheap, ubiquitous hardware
- Indoor and outdoor coverage
- Privacy observant



Downtown Seattle

Contributions

- Gaussian process + signal strength localization not new (Schwaighofer, et al. 2003)
- High accuracy Wifi localization (RSS 2006):
  - Hybrid graph-based free-space model
  - Custom kernels for Wifi
  - Robust handling of sparse training data

Outline

- Motivation
- GP for Localization
  - Introduction
  - Kernel Selection
  - Results
- GP for SLAM

Wifi Localization

We wish to model:

$P(z|x)$

where:

z = measurement

x = location

Measurement is signal strengths from visible access points:
<A=-80 B=-59 C=-26>

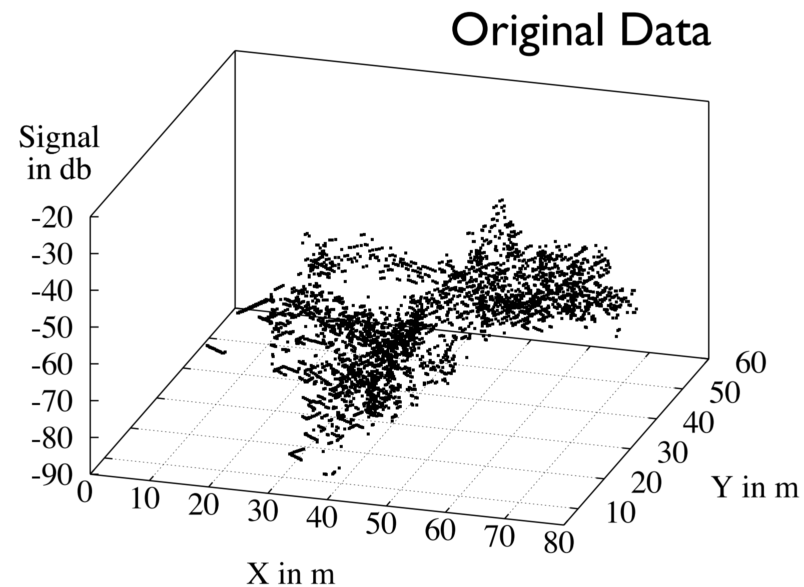
Existing Techniques

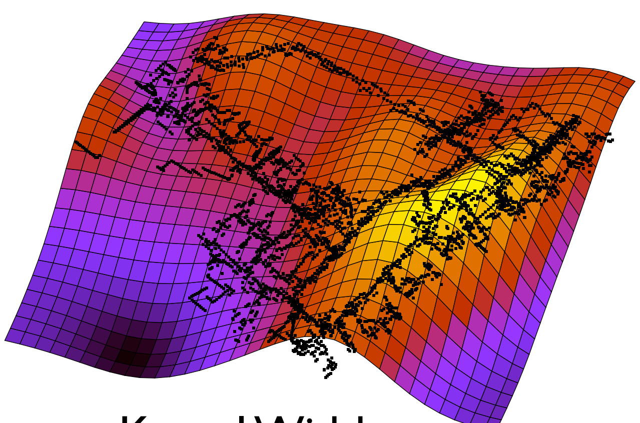
- Centroid: Given known AP locations, localize to centroid of currently visible APs
- Propagation: Attempt to model signal strength wrt. AP location, walls, furniture
- Fingerprint: Record signal strength at all points of interest
- Advanced: Hybrid models

Gaussian Processes

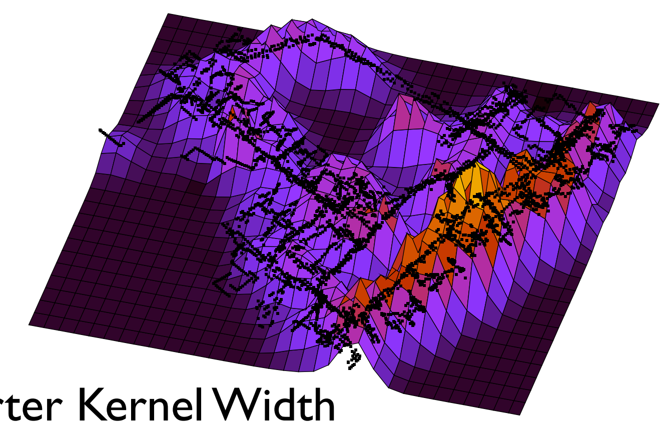
- Combines the strengths of previous techniques in one model:
  - Continuous: does not require discrete input space
  - Accurate: correct handling of uncertainty
  - Efficient: model parameter estimation

Gaussian Kernel





Longer Kernel Width



Shorter Kernel Width

Different Kernels

- Dimensional kernel: a separate Gaussian kernel each maintained for each x,y,z dim
- AP distance kernel: difference in radial distance from the access point
- Fisher kernel: includes underlying generative model of input space appropriate to Wifi

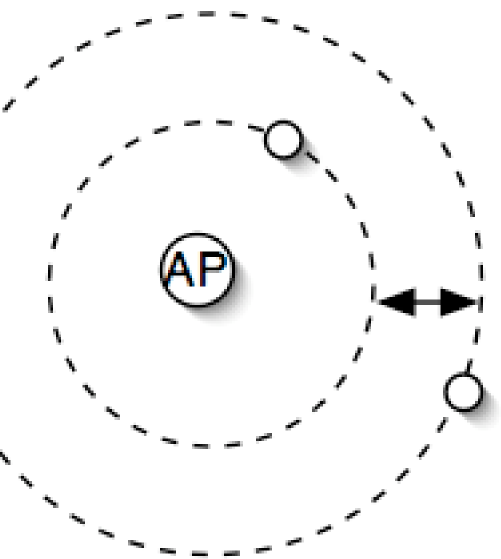
Dimensional Kernel

- Model each cartesian dimension with a separate Gaussian

$$k(p,q) = \alpha_x^2 \exp\left(-\frac{||p_x - q_x||^2}{2\sigma_x^2}\right) + \alpha_y^2 \exp\left(-\frac{||p_y - q_y||^2}{2\sigma_y^2}\right) + \alpha_z^2 \exp\left(-\frac{||p_z - q_z||^2}{2\sigma_z^2}\right)$$

- Shorter kernel width in Z dimension reflects propagation through floors

AP Distance Kernel



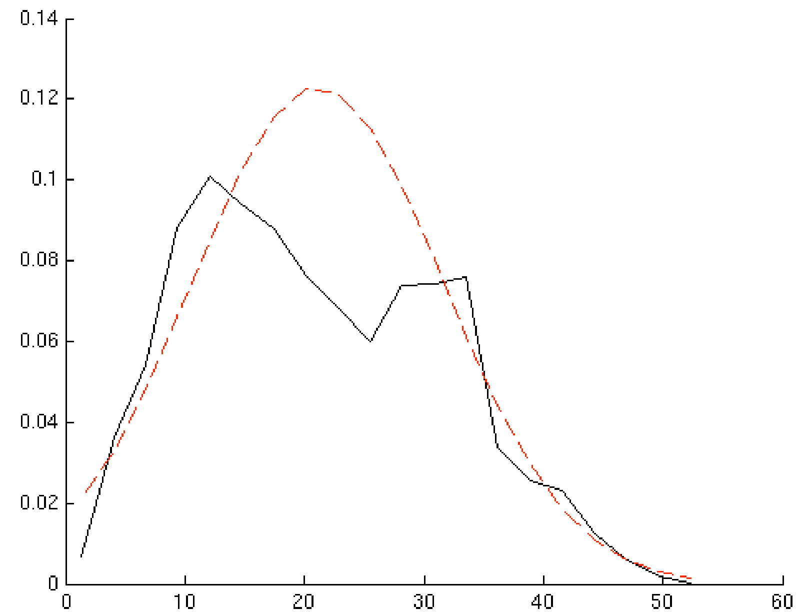
- Use difference in distance from access point of readings

$$k(x_p, x_q) = \exp\left(-\frac{(||x_p - x_{AP}|| - ||x_q - x_{AP}||)^2}{2\sigma^2}\right)$$

- Captures potential radial symmetry around the signal source
- Useful against sparse training data?

Fisher Kernel

- Incorporates a generative model of P(x) into the discriminative GP classifier
- For Wifi, we choose x as distance from the AP and model P(x) as a Gaussian



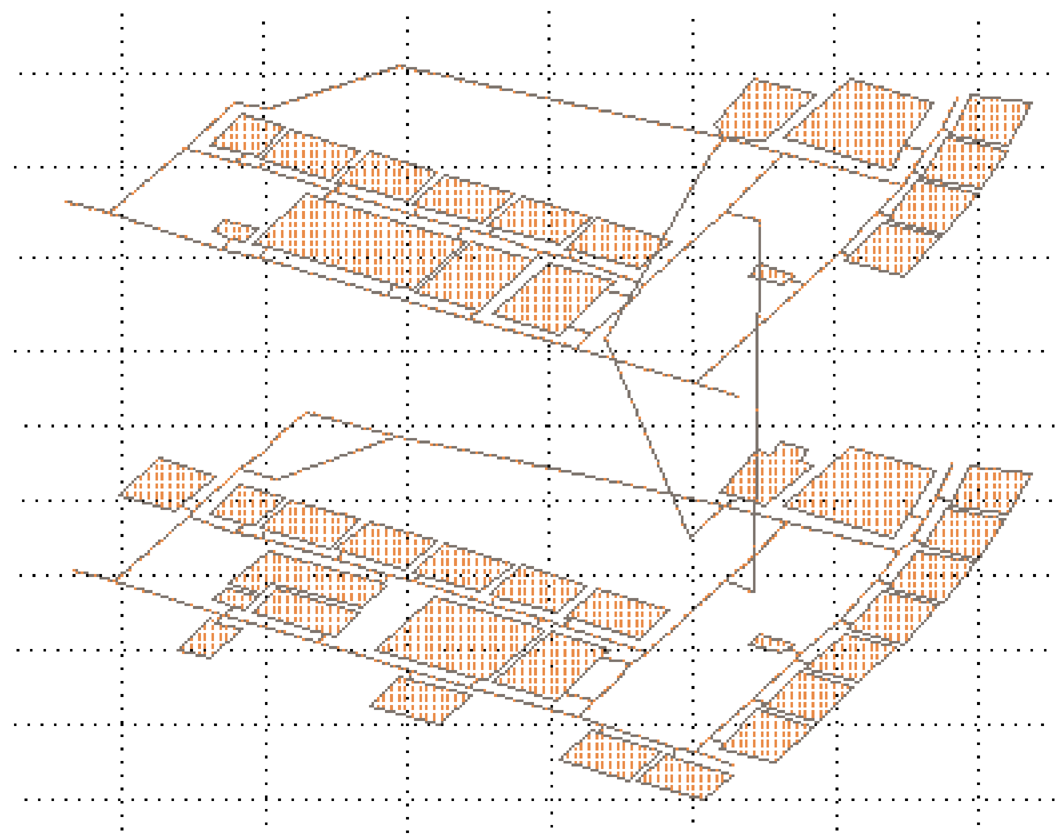
Reading likelihood vs. distance from AP

AP Location

- Kernels require location of access point
- Assume a simple linear propagation model
- Optimize AP location by minimizing difference of model vs (x_i, y_i) training pairs
  - $$f = \sum_{i=1}^{n} (y_i - m||x_i - x_{AP}|| - b)^2$$
- b = max signal strength right at the AP
- m = a negative drop-off slope

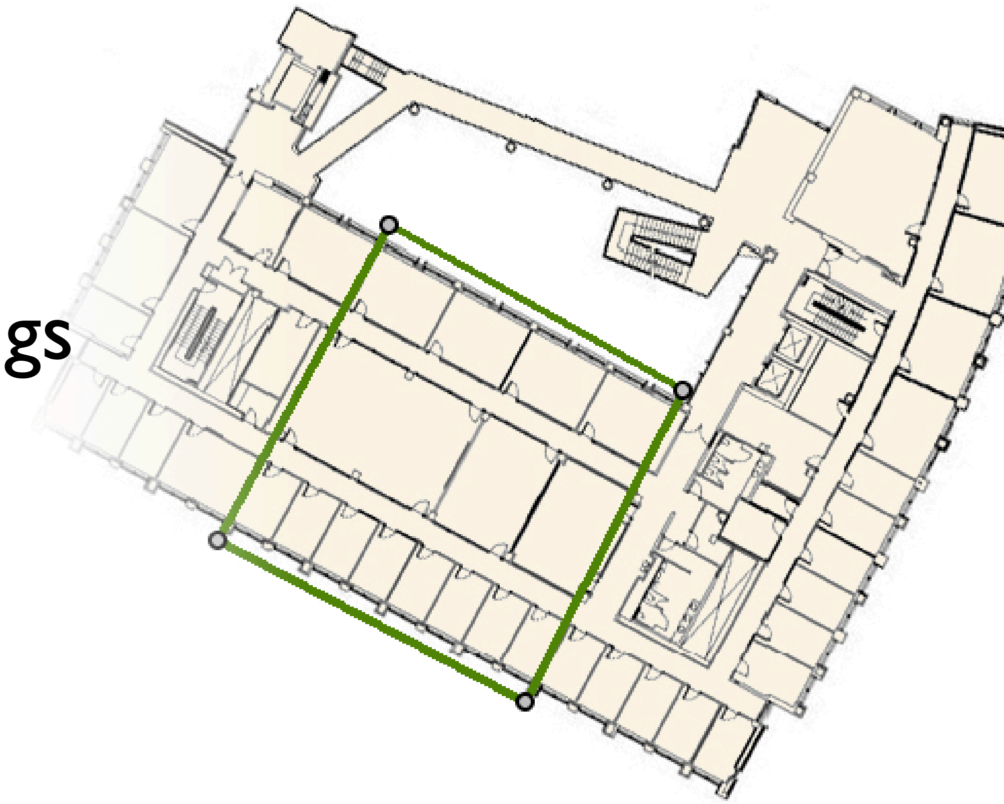
Wifi Localization

- Model each Wifi AP with a single GP
- Model building as a graph
  - Edges for hallways
  - Polygons for free space
- Particle filter for localization



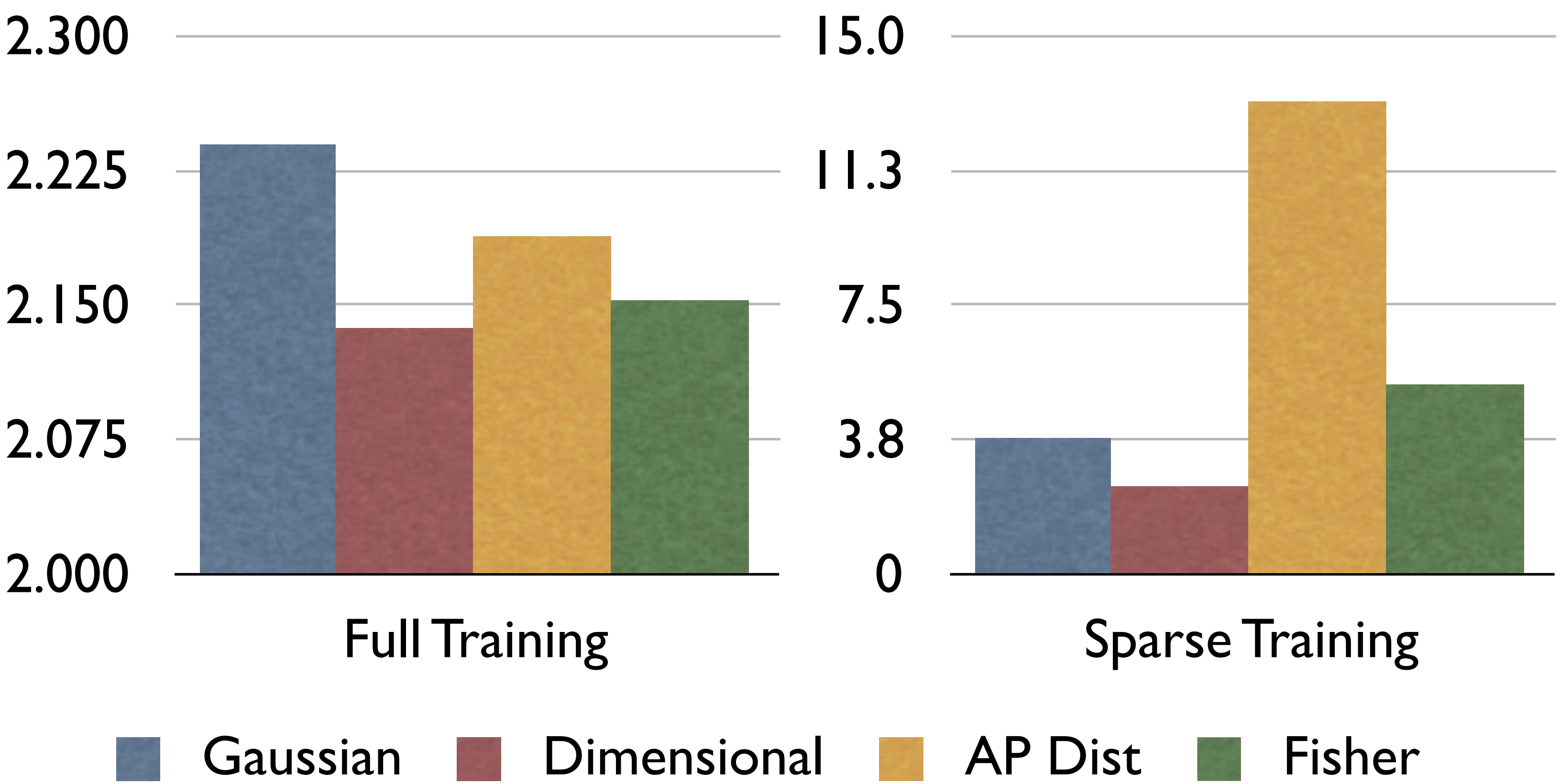
Experiments

- Training:
  - Full data: all readings
  - Sparse data: only readings outside region
- Test: 10 traces spanning hallways, offices, stairs, elevators



Kernel Results

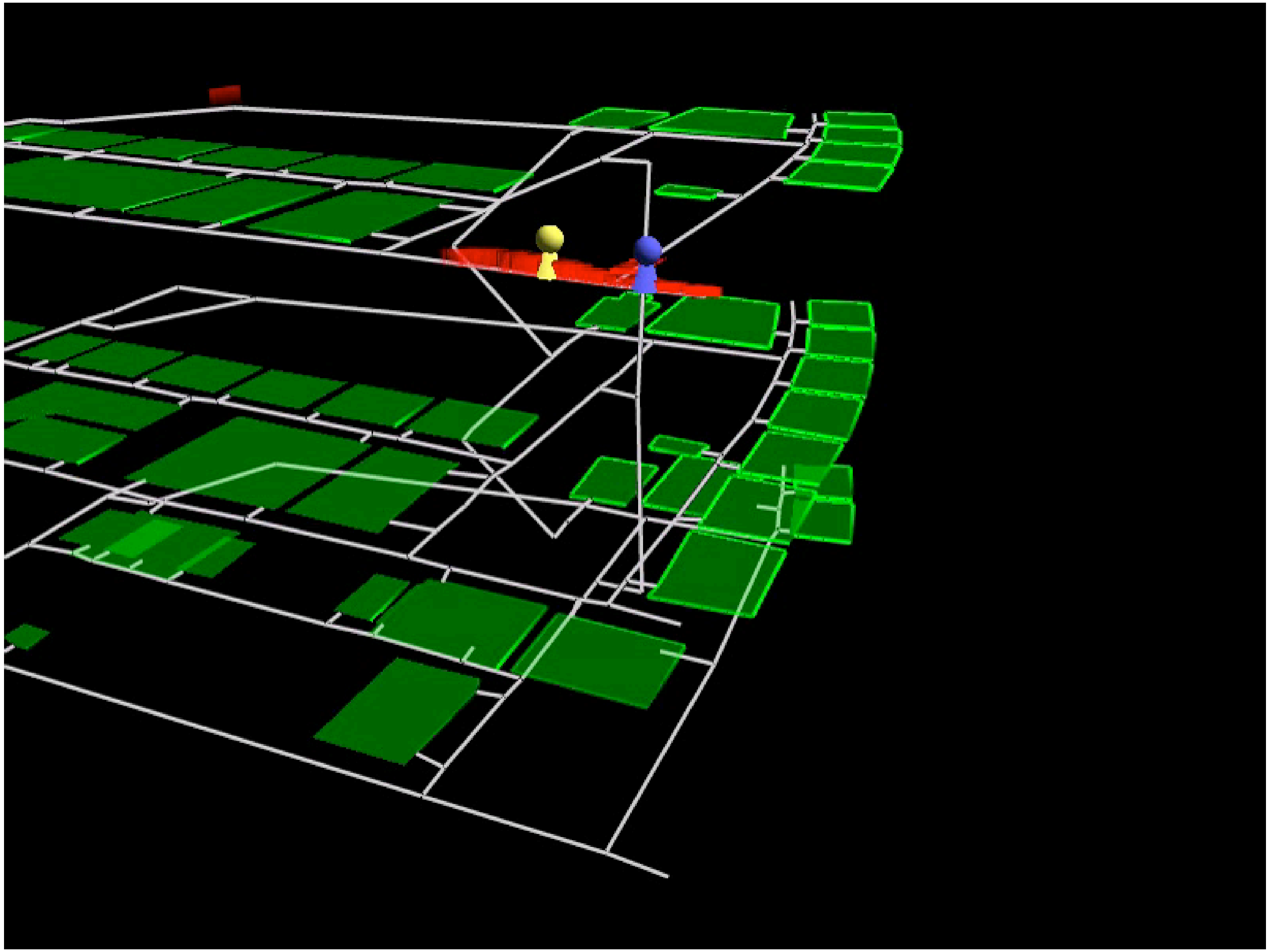
Average Localization Error (in meters)



Localization Results

- Our best-case results:
  - Online: 2.12 meters
  - Offline: 1.69 meters
  - Room classification: 80% correct
- Compared to other methods:
  - 1.8 meters [Letchner] - Hallway only
  - 2.1 meters [Haeberlen] - No extrapolation

Demo



Outline

- Motivation
- GP for Localization
- GP for SLAM
  - GPLVM
  - Dynamics Model
  - Results

Wifi SLAM

- Localization model requires labeled training data
- Can we build this model without a map?
  - Simultaneous localization and mapping (SLAM)

Wifi SLAM

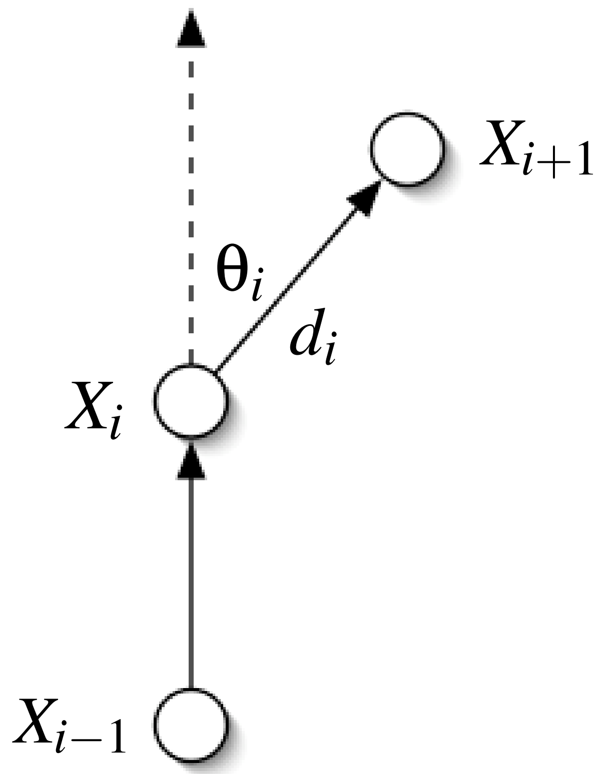
- We've already solved (Y|X) for localization
- Can we solve P(X|Y)?
  - Gaussian Process Latent Variable Modeling (GPLVM)

Gaussian Process

- Use basic Gaussian kernel
  - fixed parameters from localization model
  - forces latent space to proper scale
- Why not advanced kernels?
  - Only working in 2D
  - Access point locations add complexity

Dynamics Model

- We consider:
  - distance between latent points d_i
  - change in orientation between points θ_i



Dynamics Model

- Probability model:
  - distance:

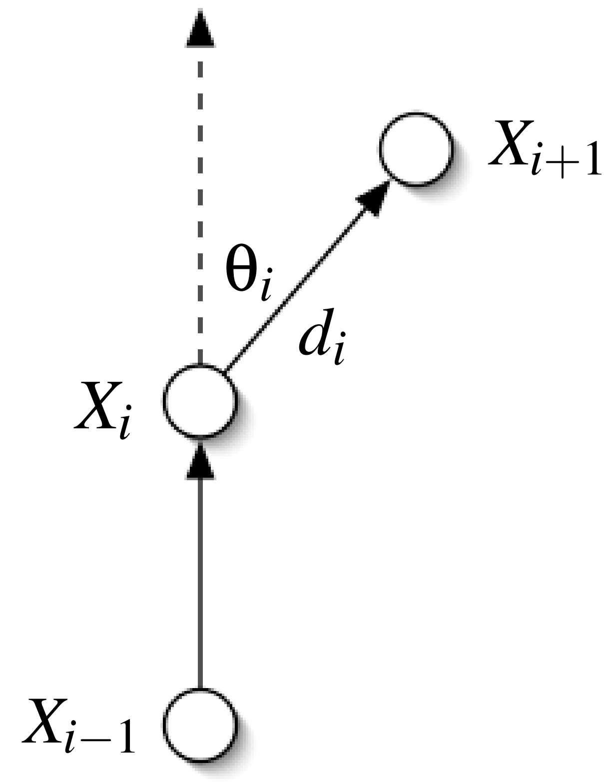
$$P(d_i|X) = \mathcal{N}(d_i, \mu_v t_i, \sigma_v t_i)$$

u_v = velocity mean σ_v = velocity sigma

  - orientation:

$$P(\theta_i|X) = \mathcal{N}(\theta_i, 0, \sigma_\theta)$$

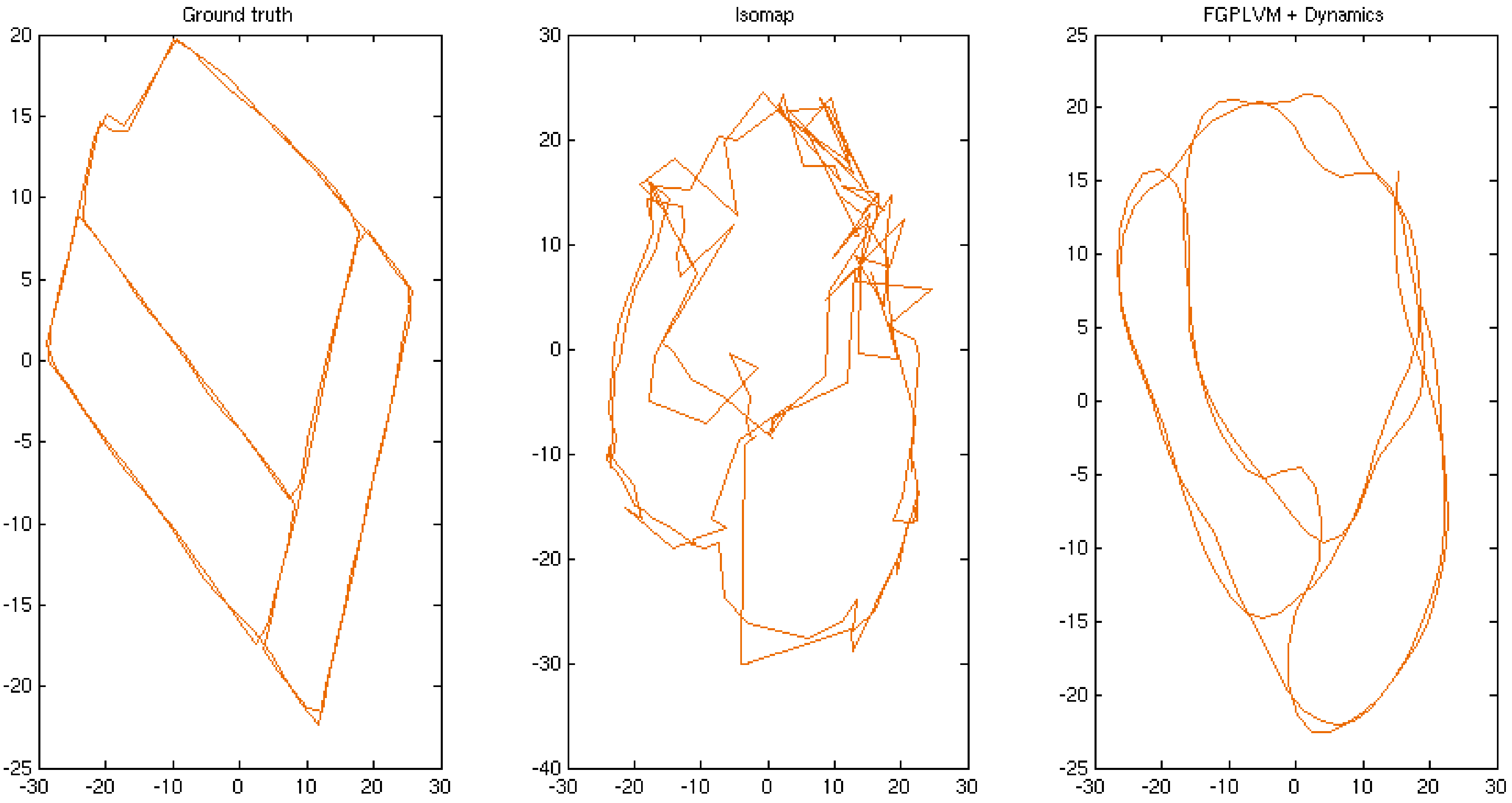
σ_θ = orientation sigma



Other Details

- Initialize with Isomap:
  - nearest-neighbor provides starting point
  - still very noisy
- FGPLVM for 1000 iterations
- Fixed parameters trained from previous localization traces

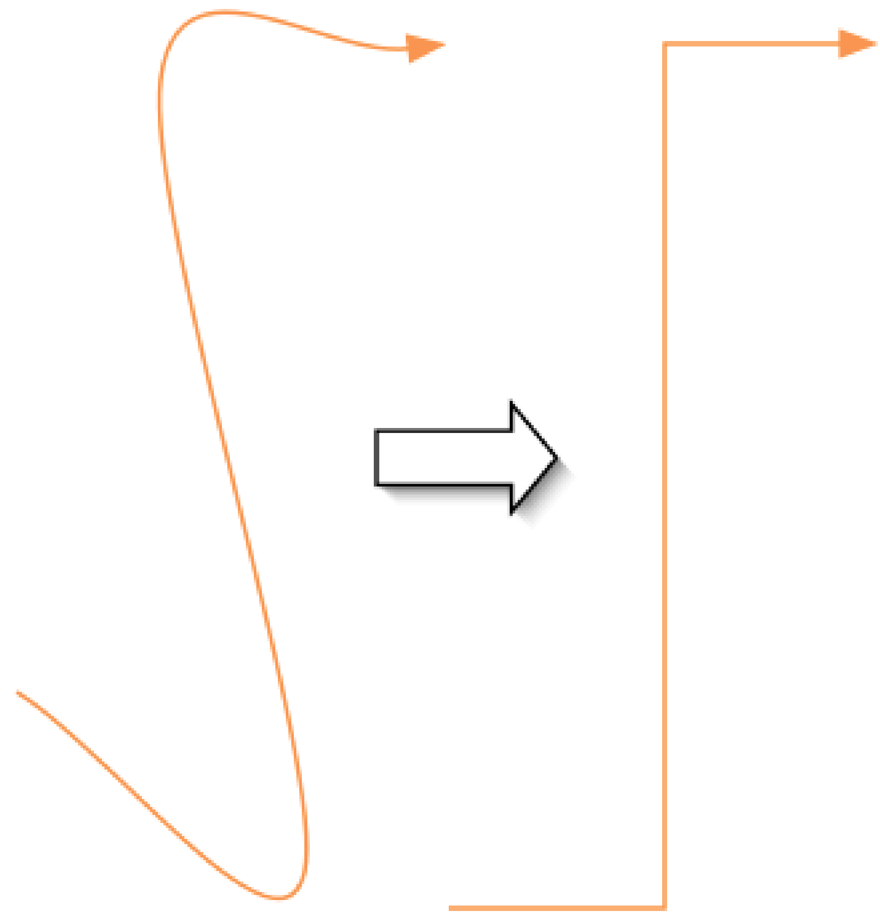
Results



University of Washington Computer Science & Engineering

Next Improvements

- More advanced dynamics models:
  - hard right angles
  - avg. hallways lengths
  - joint classification



Future Work

- Large scale Wifi localization:
  - robust indoor + outdoor
  - Social networking study with 25 users
- Continued work with Wifi SLAM:
  - Refined dynamics, odometry sensors

Questions?