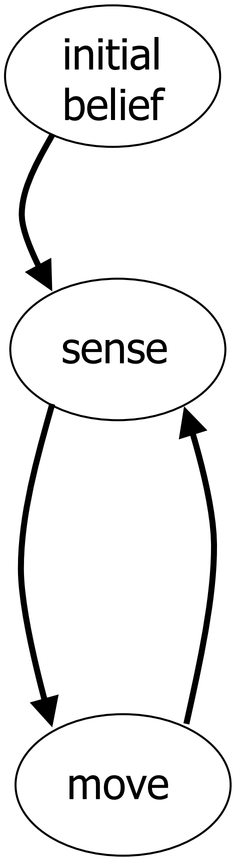
Lecture 03

SmartPhone Sensing

First assignment

- Confusion matrices
- Picture of App
- 2 page, please!
  - Don't explain things that were already explained in class.
- Code must be on phone!

Recap



current pdf (**posterior**) | perception model (sense) | pdf from last time step (prior)

$$p(X_k \mid Z_{1:k}) = \frac{p(Z_k \mid X_k) \quad p(X_k \mid Z_{1:k-1})}{p(Z_k \mid Z_{1:k-1})}$$

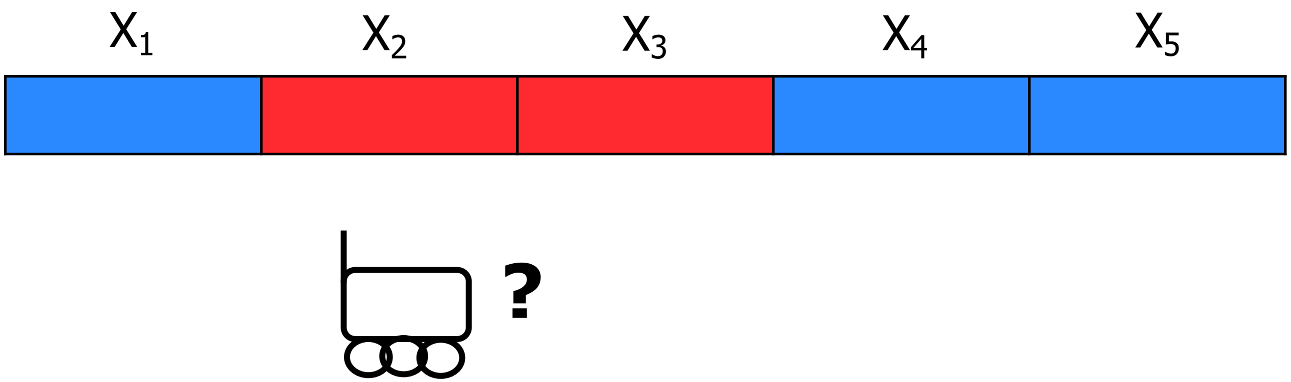
normalization

current pdf (**posterior**) | motion model (move) | pdf from last time step (prior)

$$p(X_k \mid Z_{1:k-1}) = \int p(X_k \mid X_{k-1}) \quad p(X_{k-1} \mid Z_{1:k-1})\, dX_{k-1}$$

Initial belief

a robot ***has a map*** of the grid below, but it doesn't know where it is, what is its initial belief?



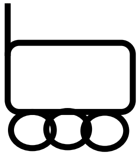
Sense

perception model

p(z="red" | Xi is "red") = 0.6 p(z="blue" | Xi is "blue") = 0.8

p(z="blue" | Xi is "red") = 0.4 p(z="red" | Xi is "blue") = 0.2

X_1	X_2	X_3	X_4	X_5
0.2	**0.2**	**0.2**	**0.2**	**0.2**



Exercise: 'Xi' represents cell i.
'z' is the sensing measurement

Question: the observation z is "red",
what is the new belief?

Sense

perception model
p(z="red" | Xi is "red") = 0.6
p(z="blue" | Xi is "red") = 0.4
p(z="blue" | Xi is "blue") = 0.8
p(z="red" | Xi is "blue") = 0.2

0.2	**0.2**	**0.2**	**0.2**	**0.2**	p(x) [prior]
0.2	0.6	0.6	0.2	0.2	p(z|x)
0.04	0.12	0.12	0.04	0.04	p(z|x)*p(x)
0.04/0.36	0.12/0.36	0.12/0.36	0.04/0.36	0.04/0.36	normalize
1/9	**1/3**	**1/3**	**1/9**	**1/9**	**p(x)** [posterior]

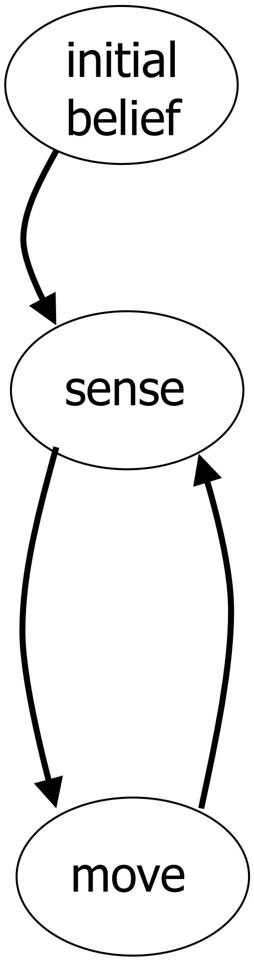
current pdf (**posterior**) — perception model (sense) — pdf from last time step (prior)

$$p(X_k \mid Z_{1:k}) = \frac{p(Z_k \mid X_k)\, p(X_k \mid Z_{1:k-1})}{p(Z_k \mid Z_{1:k-1})}$$

normalization

You need a good training phase to build your "perception model"
This is key!

Now let's look at the movement part



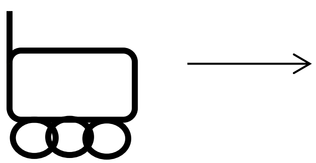
current pdf (**posterior**) | motion model (move) | pdf from last time step (prior)

$$p(X_i^t \mid X_j^{t-1}) = \Sigma \; p(X_i \mid X_j) \; p(X_j^{t-1})$$

Movement Model

movement model	$p(X_i \mid X_i) = 0.1$ $p(X_{i+1} \mid X_i) = 0.8$ $p(X_{i+2} \mid X_i) = 0.1$

X1	X2	X3	X4	X5
0	**0.5**	**0.5**	**0**	**0**

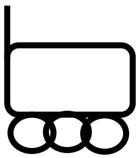


Exercise: the robot is moving right, what is the new distribution?

Move

movement model	p(Xi | Xi) = 0.1 p(Xi+1 | Xi) = 0.8 p(Xi+2 | Xi) = 0.1

X1	X2	X3	X4	X5
0	**0.5**	**0.5**	**0**	**0**
0	0.05	0.45	0.45	0.05



When using RF, you don't want to see people teletransporting in your localization app. Movements models (any model) help a ton!

Congratulations

you can now do discrete localization!!!

let’s see how this can be used with phones

Paper 1: The Horus WLAN Location Determination System

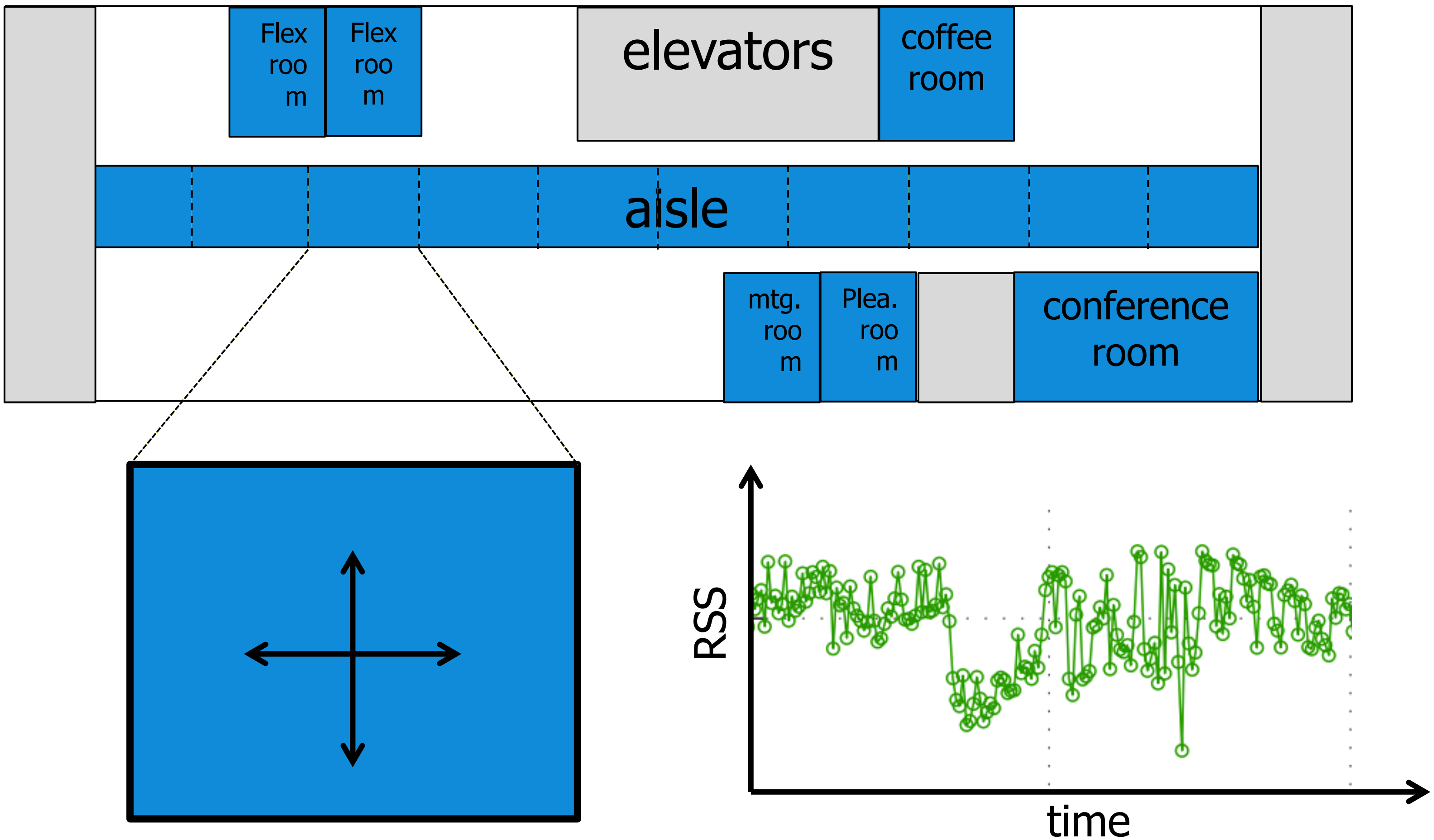
http://www.cs.umd.edu/~moustafa/papers/horus_usenix.pdf

Paper2: Practical Robust Localization over Large-Scale 802.11 Wireless Networks

http://www.kavrakilab.rice.edu/sites/default/files/mobicom2004.pdf

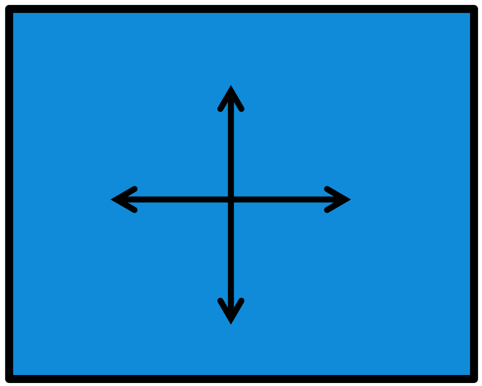
STEP BY STEP: BAYESIAN LOCALIZATION

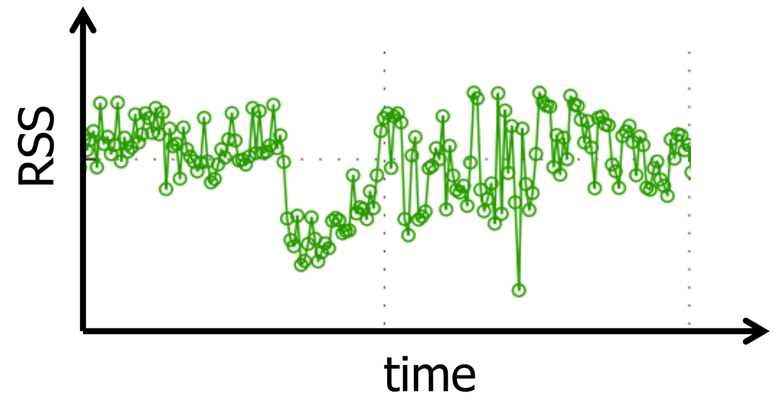
Step 1) Get RSS signals for each cell



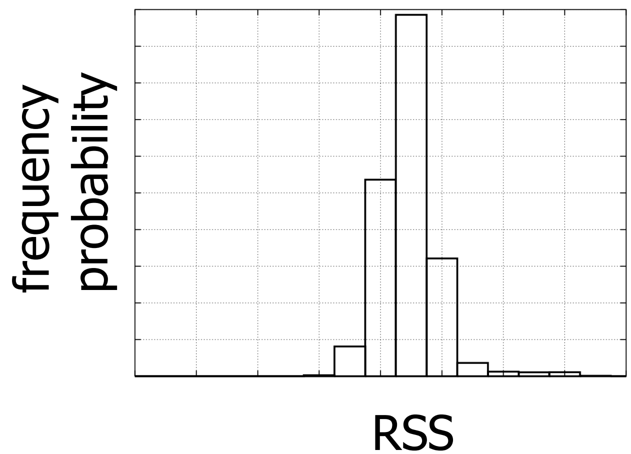
position yourself in the center of each cell to gather RSS data

Step 2) Process signal & get histogram

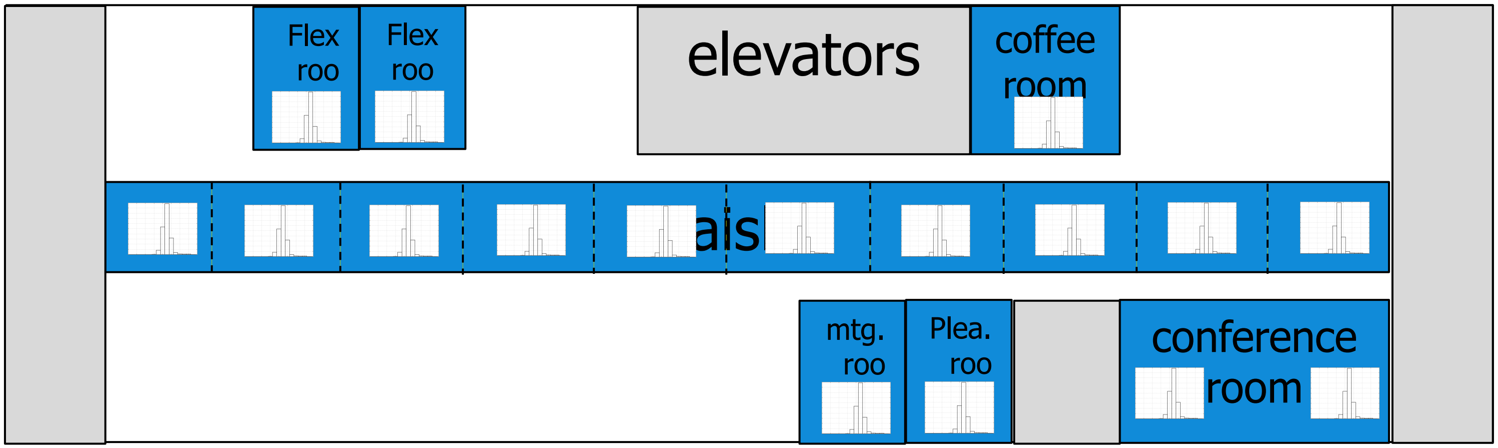




Sub-steps:
2.1) filter signal
2.2) get features (**RSS or avg RSS**)
2.3) get histogram
2.4) get pmf



Radio Map

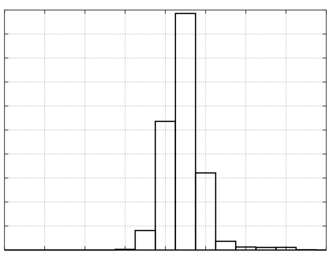


- Each cell has now a probability mass function (pmf)
- The higher the difference among pmf's the better
- Notice that the big room has two cells.

Step 3) Store data in phone

RSS values [0,255]

pmf of cells



	0	…					…	255
C1								
			0.1	0.2	0.5	0.2		
C17								

$\Sigma pi = 1$

one table per access point

localization won’t be trained on the spot

Considering the table below and given that a user measures an rss value of *r*, where is the user more likely to be?

RSS values [0,255]

pmf of cells

	…	r-2	r-1	r	r+1	r+2	r+3	…
C1		0.3	0.4	0.2	0.1			
C2		0.1	0.2	0.4	0.2	0.1		
C3			0.5		0.5			
C4			0.2	0.6	0.2			
C5			0.1	0.2	0.5	0.2		
C6				0.3	0.4	0.3		
C7			1.0					

$\Sigma pi = 1$

one table per access point

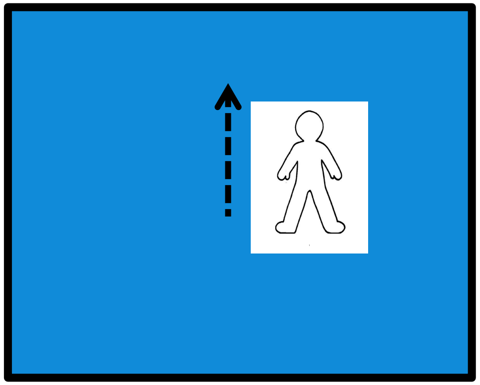
localization won't be trained on the spot

We have the radio map in the phone …

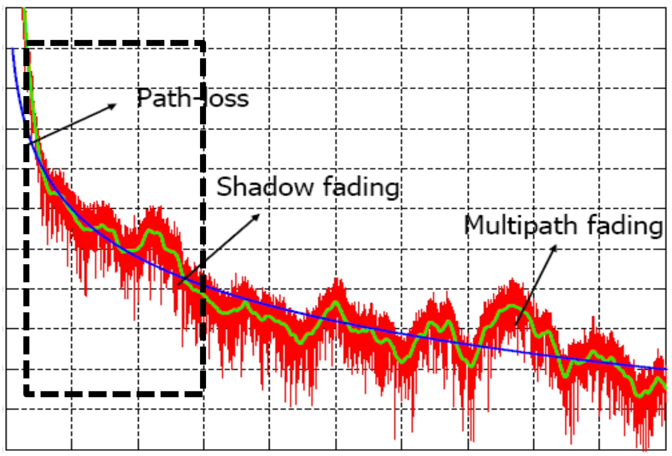
now, let’s localize

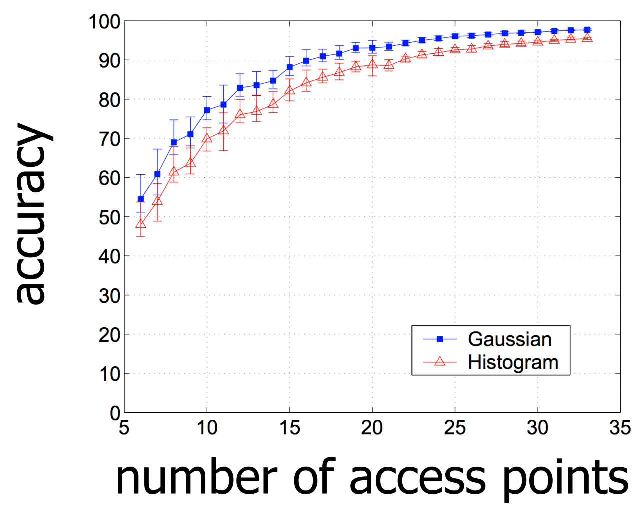
Step 4) Get testing data

- Start with initial belief
- Do WiFi scan
- Sort Access Points in decreasing order based on RSS
- Choose highest RSS, then second, etc



We will test only one direction





Step 5) Apply Bayes

probability I am in cell *i* given that I got an RSS measurement *r* from access point *j*:

$$p\,(cell_i\,/\,rss_j^{\,r}) \;=\; p(cell_i)\; p\,(rss_j^{\,r}/cell_i)\;\;/\;\;p\,(rss_j^{\,r})$$

posterior
C1
Ci
C17

=

prior
C1
Ci
C17

x

pmf of cells / **RSS values for AP j**

	0	...			r		...	255
C1								
Ci			0.1	0.2	0.5	0.2		
C17								

don't forget to normalize!

Step 6) When to stop iterations?

Step 6) When to stop iterations?

- No clear answer
- At every step you can update *prior* with
  - data from other access points
  - new scans
- Stop when you
  - pass a threshold: say 0.95
  - reach `steady state': oscillation around a max p