Smartphone Mobile Computing

CSEP590B/F Winter 2011 (first offering)

3rd Lecture, 24 January 2011

Instructor:

Gaetano Borriello

Teaching Assistants:

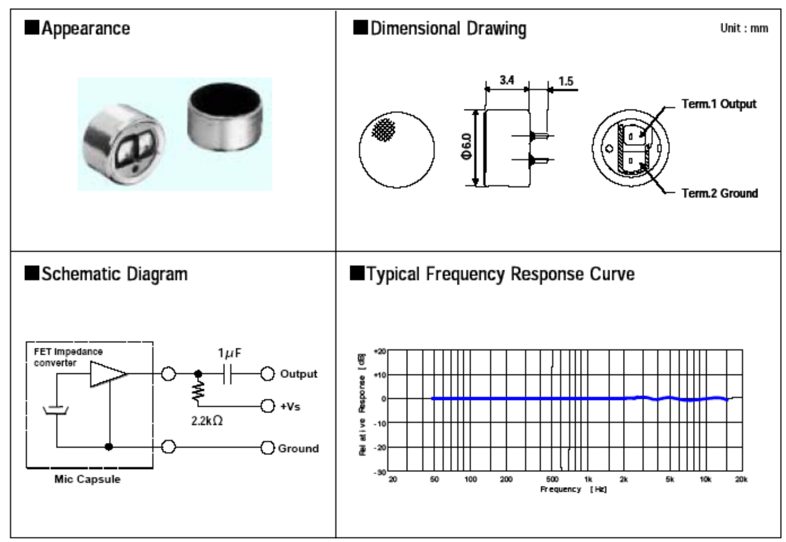
Temitope Oluwafemi and Cary Anderson

Overview for Today

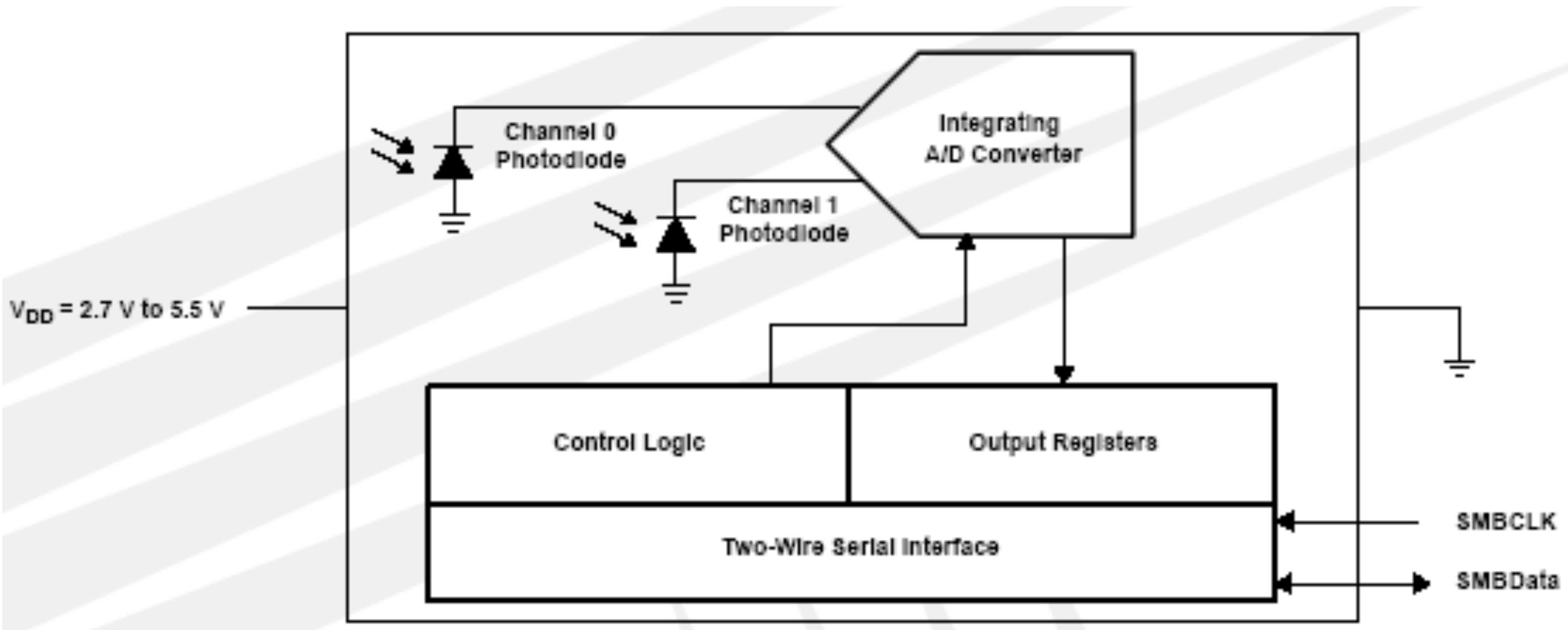
- Location systems – Jeff Hightower (Intel Seattle)
- Q/A on location systems
- Sensors on smartphones and how they are integrated
- Uses of the sensors (outside their limited intended function)
- ML applied to sensor data for determining user context

- Group projects feedback

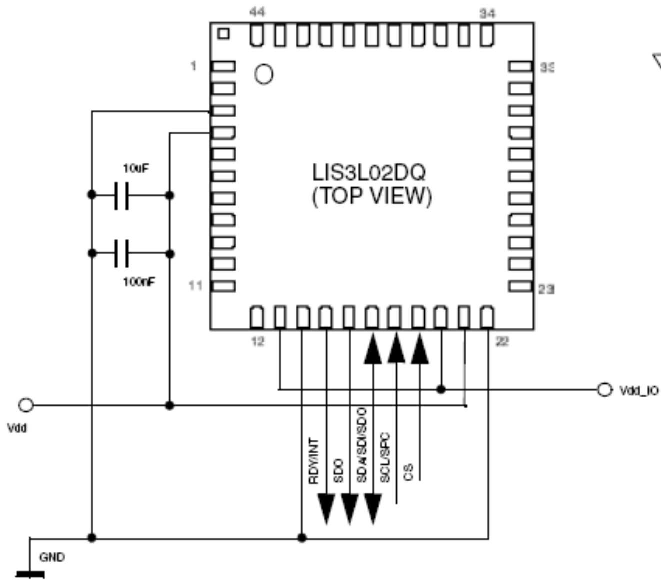
Microphone/speaker

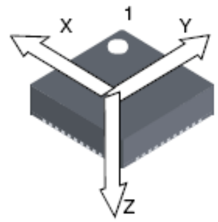


Phototransistor

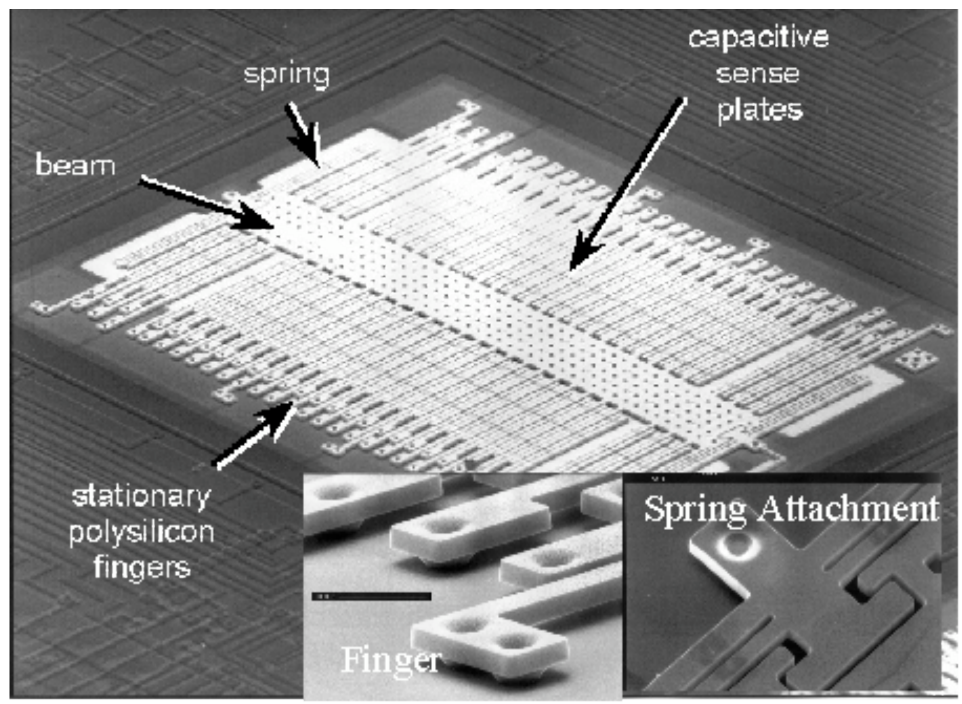


Accelerometer





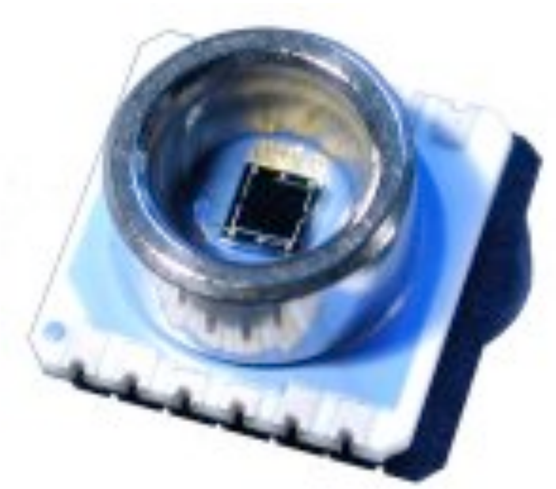
DIRECTION OF THE DETECTABLE ACCELERATIONS

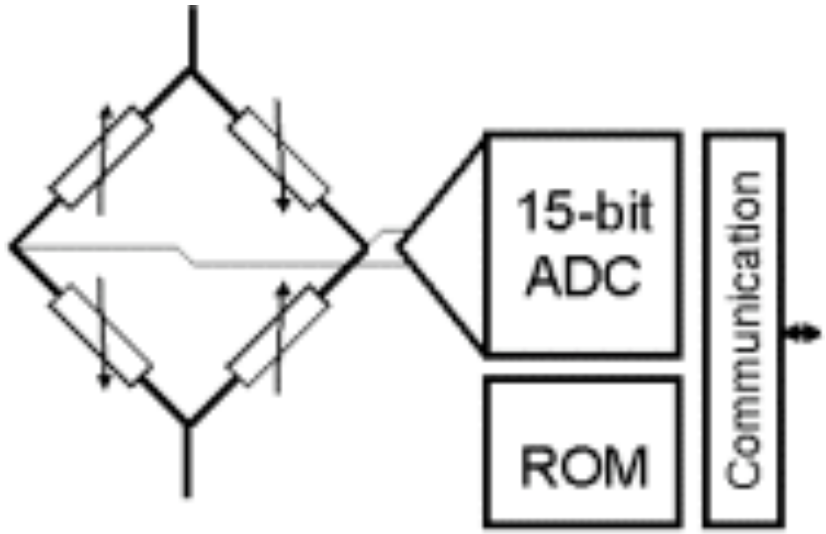


Compass

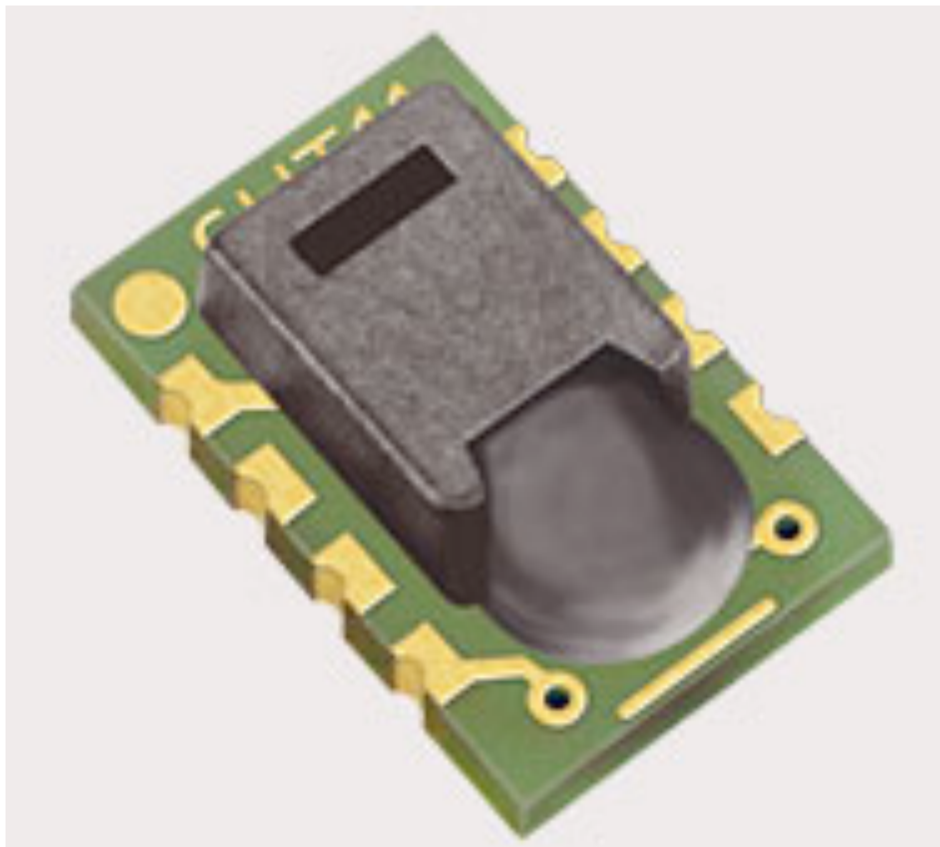

LSM303DLH
FIRST AVE
300 ft/North East
Total: 35 mi
Main St
First Ave
5
Main St
Nav
Traffic
Summary
Search Along
Phone
Mail
Web
Music

Barometer/Temperature





Humidity/Temperature



Camera

Touch screen

Communication to processors

- **GPIO (general-purpose I/O)**
  - Direct manipulation of pins of microcontroller
  - Pulse-width modulation (duty-cycle % is the value)
- **Serial connections**
  - SPI (Motorola) – serial peripheral interface bus – is most common
  - commonly used for microcontroller to peripheral connections, 10Mbits/sec
  - registers in device written and read in shift register fashion
- **Analog**
  - A-to-D conversion to 8 to 16 bit resolution
  - battery-level sensing
- **Parallel units to main processor**
  - interrupt routines inform controller when operation is complete
  - results in internal dedicated registers

Communication to processors

- **More complex devices (e.g., touch screens, cameras)**
  - often memory-mapped (e.g., screens, keyboards)
  - include their own separate microcontroller
    - to manage events
    - to refresh data in registers and on-screen
    - to control CCD array and lens for auto-focus

- **Typical smartphone has several separate processors**
  - screen (display and touch)
  - camera (auto-focus, CCD)
  - cellular interface (dsp for physical channel to packet)
  - wi-fi interface (dsp for physical channel all the way to TCP/IP protocol)
  - SIM module

Typical sensors on smart phones

- Buttons and switches
- IR proximity sensors
- Touch screen
- Camera(s)
- Accelerometer
- Compass/magnetometer
- Microphone
- Battery level

Typical actuators on smartphones

- Speaker
- Vibration engine
- Display
- LEDs (e.g., flash, keys)

Context-aware applications [Schilit/Adams/Want 94]

- **Applications that consider a user's**
  - activity
  - social situation
  - location
  - other …
- **and adapt their behavior accordingly**
  - prompt the user
  - trigger other applications
  - change their display modality
- **They may also consider past and (likely) future context**

Important application space – medically significant applications

- **Health and fitness**
  - motivating healthy behavior
  - understanding behavior traits that lead to obesity
- **Long-term behavior monitoring**
  - identify shifts in activity and interaction trends
  - make caregivers even more effective
- **Predict impending medical events**
  - detect behaviors that can predict certain medical emergencies/ conditions – e.g. chronic pulmonary disease
- **Assistance for people with cognitive dysfunction**
  - assistance in managing self-stimulatory behavior

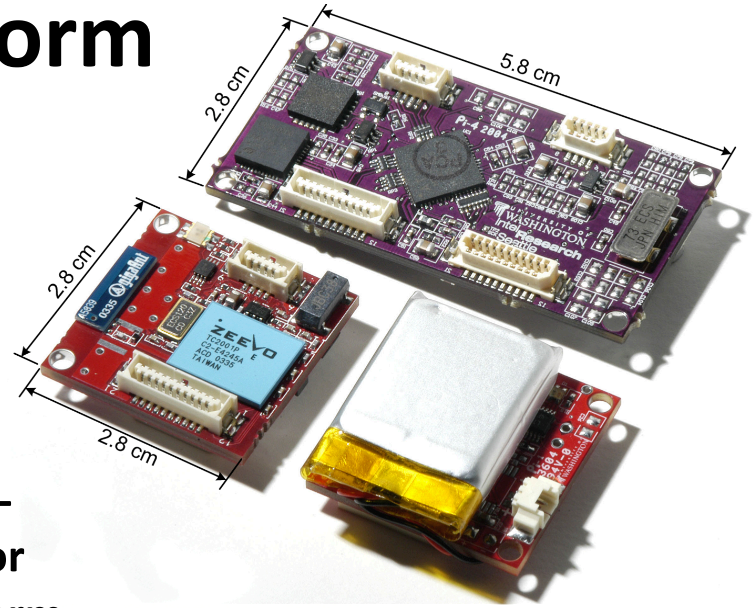
Ideal device

- **Practical, personal device**
  - cell-phone form-factor
  - full-day functionality
  - easily portable and/or wearable
- **Gracefully adaptable to available resources**
  - recognition in real-time
  - enhanced journal-keeping
  - connectivity with cloud and other sensors in environment

- **Today's smartphone already meets these needs**
  - power still an issue
  - privacy a big concern

Experimental sensor platform

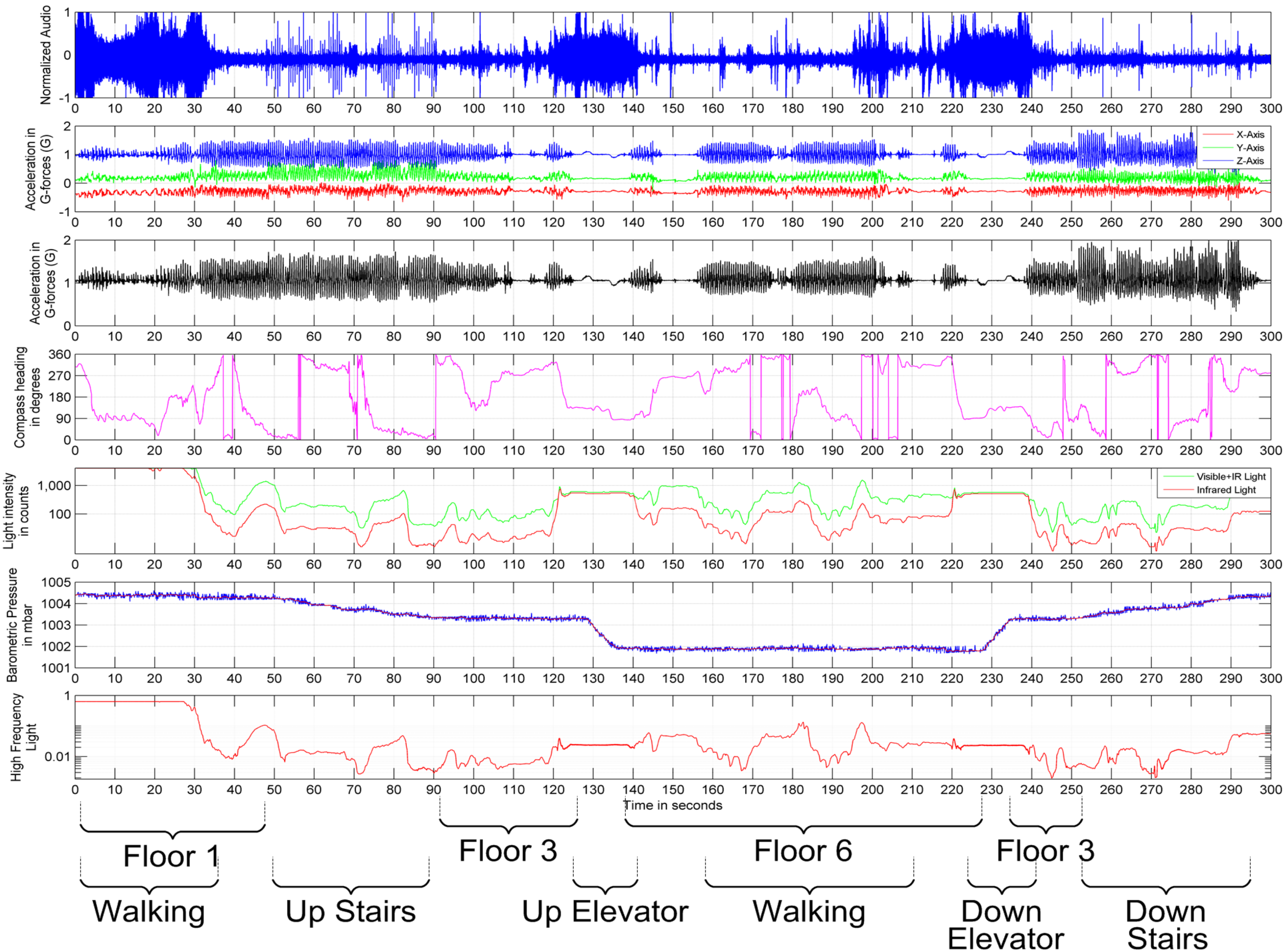
- **Seven types of sensors**
- **Connect in two modes**
  - wired over USB
  - wireless over Bluetooth
- **Collect data on phones**
  - arbitrary sample rates
- **Low power**
  - Runs for a day on small battery
- **Light-weight**
  - 65 grams, including BT base 121g
- **Form factor**
  - small, but eventually integrate into phone



MSP: Multi-Sensor Platform

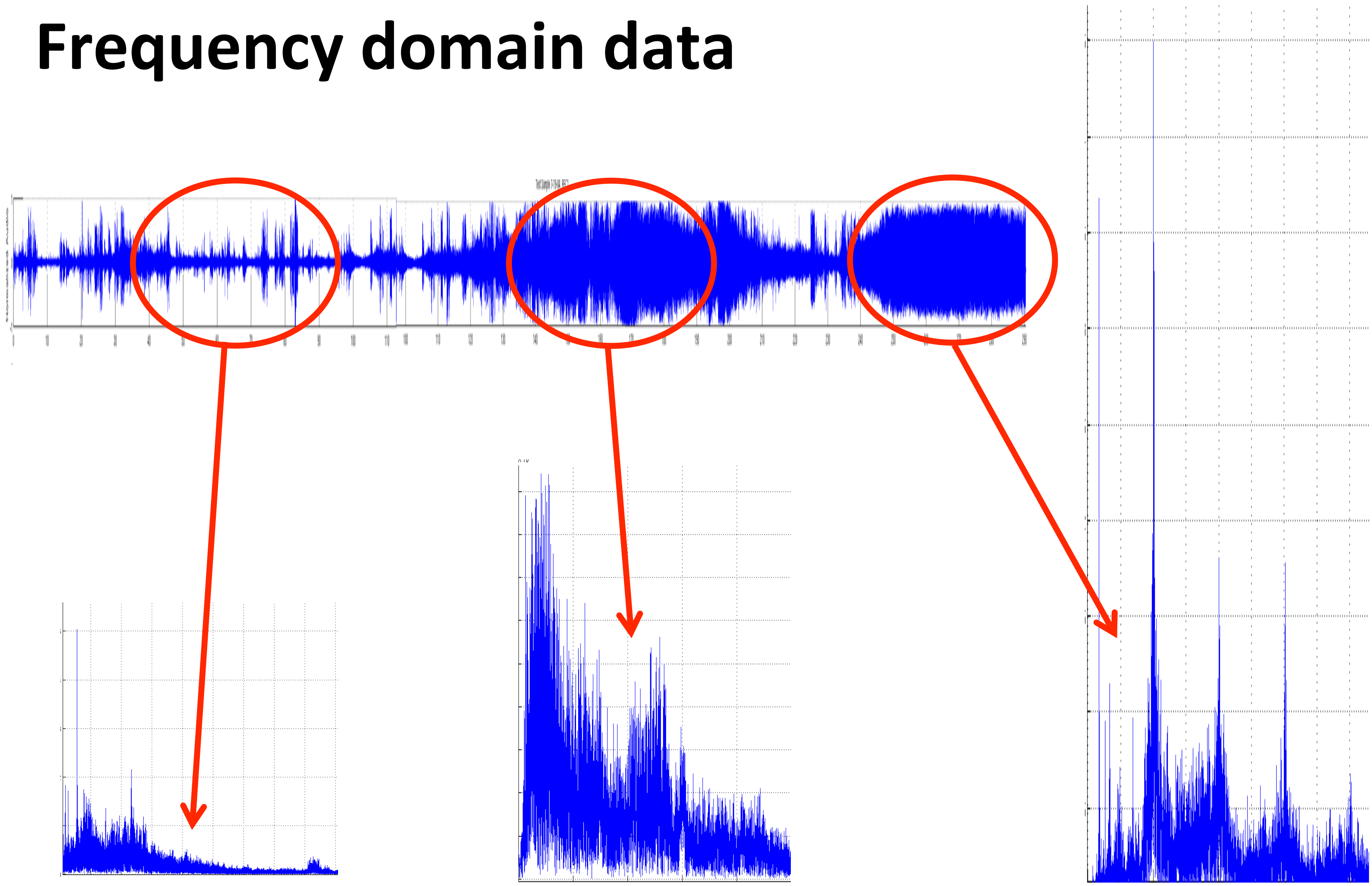
Sampling frequency

- **Sensor data collected**
  - Audio (~15kHz)
  - Visible Light (~550Hz)
  - 3-Axis Acceleration (~550Hz)
  - 2-Axis Digital Compass (30Hz)
  - Barometric Pressure (14Hz)
  - Ambient IR Light (5Hz)
  - Ambient Visible Light (5Hz)
  - Humidity (2Hz)
  - Temperature x2 (2 & 14Hz)



Normalized Audio
Acceleration in G-forces (G)
X-Axis
Y-Axis
Z-Axis
Acceleration in G-forces (G)
Compass heading in degrees
Light intensity in counts
Visible+IR Light
Infrared Light
Barometric Pressure in mbar
High Frequency Light
Time in seconds
Floor 1
Floor 3
Floor 6
Floor 3
Walking
Up Stairs
Up Elevator
Walking
Down Elevator
Down Stairs

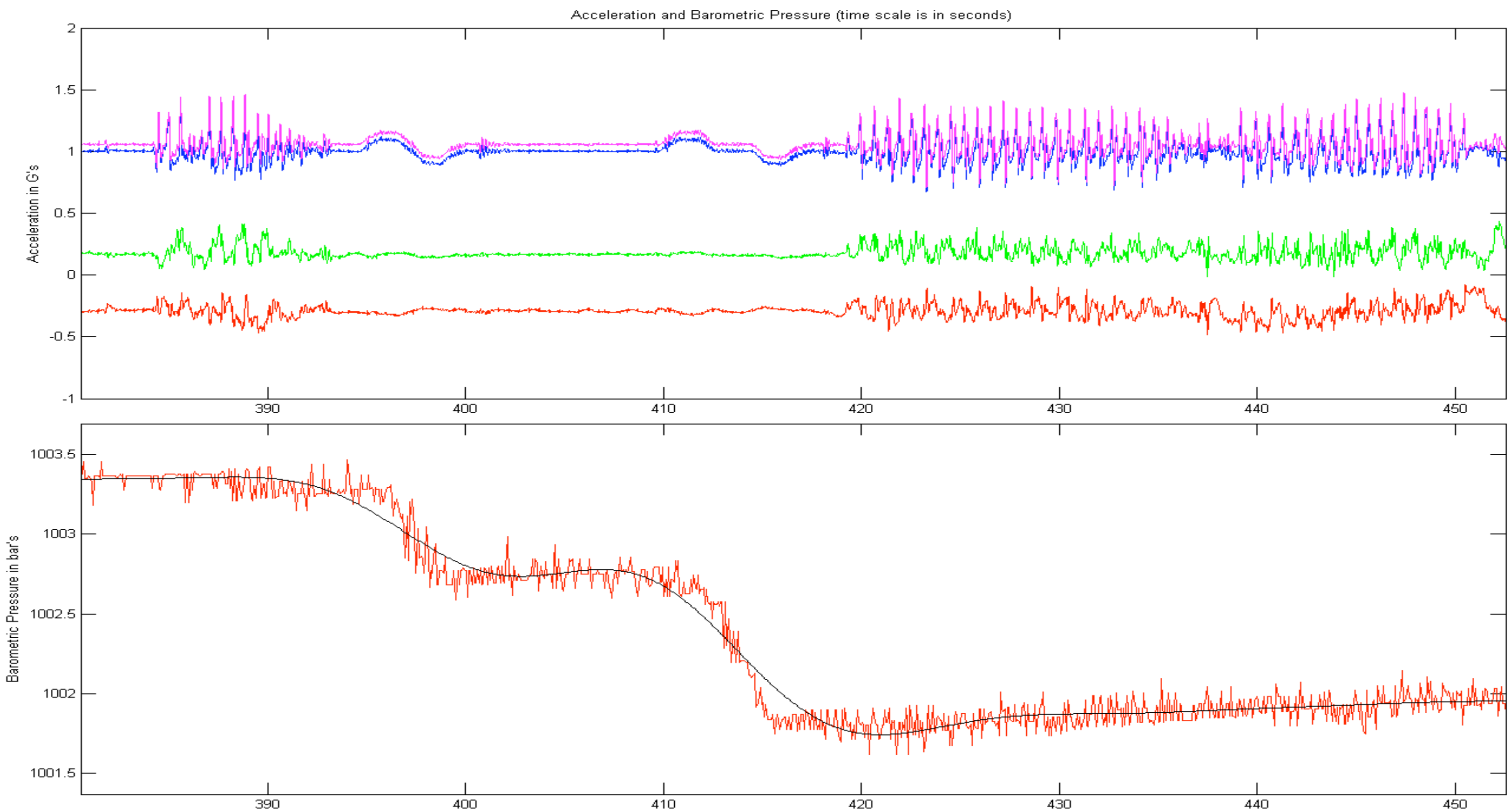
Frequency domain data



Basic Features

- Mean
- Variance
- Derivative
- Integral
- Correlation
- Fast Fourier Transform
- Matched filters
- Edit distance

Compound features and correlation



Mean: E(x) = μ

- **Average value of a signal**
  - over how large a window of data?
  - how do we move to the next window?
    - disjoint window? – new value every window time
    - overlapping windows? – new value every overlap time
    - sliding window? – new value of mean for every sample

- **Creates a new data stream**
  - same rate (sliding window)
  - slower rate (overlapping or disjoint windows)

- **Used to eliminate "noise"**
  - small variations in signal caused by sensor imperfections or wiring interference or sensitivity of measurement

Variance: sqrt (E($(x - \mu)^2$)) = σ

- **Standard deviation**
  - root mean square difference of signal from its mean
  - similar issues of window size and slide
- **Gives a rough estimate of "noise" level for a sensor whose value should not be changing**
  - doesn't work well if sensor "drifts"

Derivative ($\Delta x/\Delta t$) and integral ($\int x \, \Delta t$)

- **Derivative: difference in value over a difference in time**
  - window size/slide
  - on raw signal or mean (smoothed version of signal)?

- **Integral: area under curve traced out by data samples**
  - window/slide
  - rectangle methods: area = $\Delta t * x$
    - x on which end of Δt
  - trapezoidal methods: area = $\Delta t * (x_1+x_2)/2$

Correlation: $[\ E(\ (x - \mu_x)(y - \mu_y)\ )\ ] / \sigma_x * \sigma_y$

- **Measure of how closely two signals track each other**
  - offset by the mean of each signal
  - correlation ranges between -1 and 1
    - 0 – no correlation
    - -1,1 – exact match (-1 in opposite direction, 1 in the same direction)
  - if two signals are independent then correlation is likely to be close to 0, but converse is not true
  - if two signals are created by the same phenomenon then correlation should be close to 1 or -1, but converse is not true
  - window/slide

Fast Fourier transform

$$f_j = \sum_{k=0}^{n-1} x_k e^{-\frac{2\pi i}{n} jk} \qquad j = 0, \ldots, n-1$$

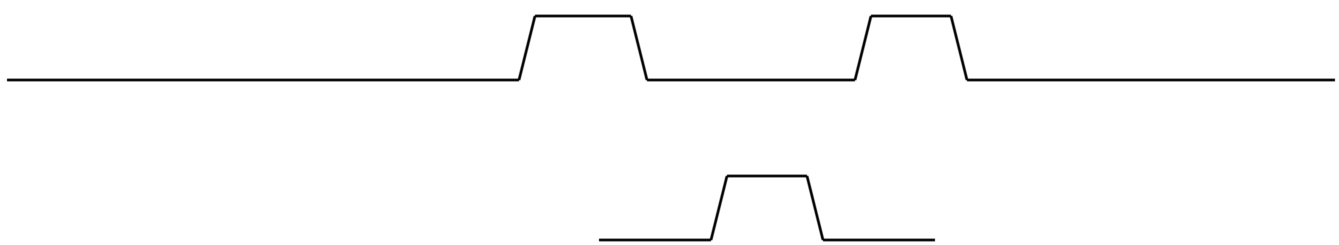
- **Extract component frequencies of signal**
  - value of each f corresponds to coefficient of that frequency
  - band-limited by Nyquist frequency
  - defined over complex signals (we use only real values)
  - simple implementation is $O(n^2)$ operations but same result can be computed in only O(n log n) operations, hence "fast", fastest when n is a power of 2
  - there is also an inverse transform to reconstruct all x_k
  - series of coefficients over time generates a spectrogram (time-varying spectrum of signal)
- **Goertzel algorithm**
  - fast implementation to determine result at only one frequency

Fast Fourier transform features

- **Linear FFT Bands**
  - sum of FFT coefficients in linearly spaced frequency bands
- **Log FFT Bands**
  - sum of FFT coefficients in logarithmically spaced frequency bands
- **Energy**
  - sum of the FFT spectrum over all bands
- **Spectral entropy**
  - measure of the distribution of the frequency components
- **Butterworth filter bands**
  - FFT bands after bandpassing (eliminating certain freq components)
- **Cepstral coefficients ("ceps" is "spec" backwards)**
  - FFT(log(FFT(x)))

Matched filter

- **Convolution of one signal with another**
  - look for a pattern (one short signal stream) in another (continuous)
  - e.g., radar pulse – look for same pulse shape in return signal
  - pattern must be a very close match in duration



Edit distance

- **Translate signal stream into a finite alphabet**
  - edit distance is defined as smallest number of changes (deletions, insertions, and/or replacements) to make one alphabet stream look like the other
  - different form of correlation or matched filters
    - not as sensitive to time dilation

Which features make the most sense?

- **Different sensors have different properties**
  - not every feature make sense for each
- **Correlations**
  - some correlations make sense others don't (independent)
  - which sensor streams are likely to be correlated
    - accelerometer?
    - light?
    - temperature/humidity?
- **Let's consider our sensors**

Accelerometer

- **Integration - dead-reckoning**
  - integral yields change in velocity
  - integral of velocity yields change in position
  - noise causes error – must estimate starting position and velocity
- **Correlations**
  - across multiple axes
- **FFT**
  - human motion typically under 20Hz (only need to sample at 40Hz)

Microphone

- **Mean**
  - ambient sound level
- **Variance**
  - noise level
- **FFT**
  - spectrum can be used as a “fingerprint”
- **Cepstral coefficients (FFT of spectrogram)**
  - useful in speech/music recognition

Light sensors

- **Mean**
  - ambient light level
- **Derivative**
  - light on / light off
  - transition to different room
  - shadow casting
- **Correlations**
  - across visible and infrared light
    - high correlation – indoor?
    - low correlation – outdoor?

Barometer

- **Derivative**
  - change in vertical position
- **Matched filter**
  - profile of elevator acceleration/deceleration
  - width of pulse varies with elevator
  - spacing of pulses indicates vertical distance

Compass

- **Edit distance**
  - alphabet: E, W, N, S, NE, SE, NW, SW, etc.
  - comparison to known patterns
    - turned around
    - went down one floor on stairways
- **Correlations**
  - barometer and compass – stairway vs. elevator

Location

- **Mean**
  - centroid of locations
- **Variance**
  - degree of movement
- **Edit distance**
  - translate to significant locations (home, work, gym, HUB, etc.)
  - look for similar transition patterns (home -> work -> home)

Some features to compute from MSP

Collect approximately 18,000 samples of data/second from which 651 features are computed

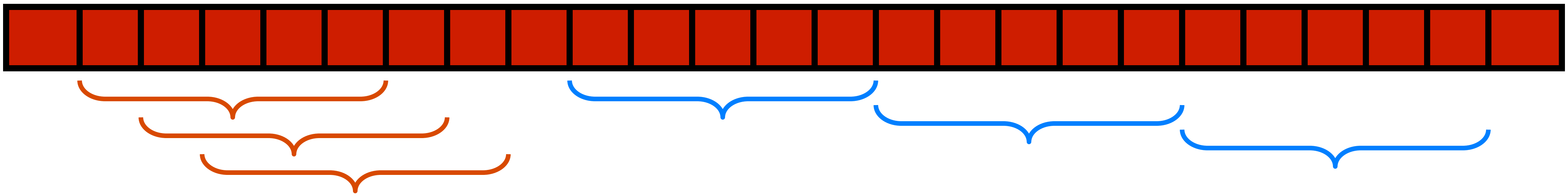
For example:

Feature	Description
Cepstral Coefficients	The FFT of the decibel FFT spectrum, that is FFT(log(FFT(x)))
Log FFT Frequency Bands	Real valued FFT values grouped into logarithmic bands
Spectral Entropy	Measure of the distribution of frequency components
Energy	The sum of the FFT spectrum
Mean	The average value of the time series
Variance	The variability of the time series
Linear FFT Frequency Bands	Real valued FFT values grouped into linear bands from 100Hz - 2kHz
Butterworth Filter Bands	The sum of band pass filtered bands from 100Hz - 2kHz
Correlation Coeffs	Correlation between axis pair, XY, XZ, YZ
Trapezoidal Integration	Integrated value of the time series over the window

Data Segmentation

- **What is a segment of data?**
  - A long enough segment of the data stream to include characteristics of the activity in feature space
  - For most human activities this ranges from 0.1 to 5.0 secs
- **What do we pick as a segment size?**
  - 0.1 may be too small – for a few steps up a stairway
  - 5.0 may be too big – for stopping while walking
- **Manual or automatic segmentation?**
  - Manual more precise but more time consuming
  - What provides ground truth for automatic methods?

Sliding Window on Data Stream



To overlap or not to overlap

- **Do a combination**
  - Small basic window size
  - Features over longer interval

- **In our case:**
  - 0.25 sec window size
  - Some features use data from 1.0 seconds
  - Do we need longer?

Examples

- **Accelerometer**
  - Samples at 549/sec
    - Can detect up to 250Hz
  - Features from humans at 0.5 to 3Hz
    - 2000 steps/mile at 3mi/hr yields 6000/3600 or ~1.5Hz for walking
    - Need to sample at least twice this rate (Nyquist criterion)
    - Need a few seconds to see walking pattern in FFT
- **Microphone**
  - Samples at 15630/sec
    - Can detect up to 8KHz
  - Features for human speech at 20Hz-20KHz
    - Fraction of second is enough to detect most specific frequencies
    - May also want to look at sequence of patterns
      - – cepstral coefficients are a simple case

Computation

- **Window size and feature length**
  - Imply computational costs
- **Compute all features for every window**
  - Every 0.25 secs (8 512-point FFTs)
  - Use up to 1 sec of data (1 15360-point FFT)

- **Smaller windows?**
- **Features over longer data streams?**

Activity data – training and test cases

Activity	Duration		Instances
Sitting	13 hrs	4 mins	81
Standing	2 hrs	6 mins	114
Walking	8 hrs	56 mins	419
Jogging		19 mins	21
Walking up stairs		39 mins	87
Walking down stairs		31 mins	69
Riding a bicycle	1 hrs	4 mins	4
Driving car	1 hrs	20 mins	3
Riding elevator down		19 mins	93
Riding elevator up		21 mins	107

Average Duration:	2 hrs	52 mins	99.8
Relevant Labeled Data:	28 hrs	39 mins	
Total Recorded Data:	37 hrs	57 mins	

Activity	Duration	Instances
Brushing Teeth	9 mins	6
Driving a car	53 mins	3
Eating	21 mins	5
Jogging	2 mins	3
Riding elevator down	10 mins	27
Riding elevator up	10 mins	26
Scrubbing Dishes	11 mins	7
Sitting	25 mins	6
Standing	34 mins	59
Vacuuming	4 mins	1
Walking	28 mins	81
Walking down stairs	4 mins	11
Walking up stairs	6 mins	14
Watching TV	14 mins	1
Working on computer	18 mins	2

Average Duration:		17 mins	16.8
Relevant Labeled Data:	4 hrs	8 mins	
Total Recorded Data:	6 hrs	10 mins	

What about other activities

- **Housework**
  - Vacuuming
  - Washing dishes
  - Dusting
  - Sweeping
  - …
- **Personal hygiene**
  - Brushing/combing hair
  - Washing face
  - Showering
  - Brushing teeth
  - …
- **Physical exertion**
  - Walking up/down hills
  - Degree of slope
  - Lifting weights
  - …
- **Entertainment**
  - Dancing
  - Watching TV
  - Attending a concert
  - Attending a party
  - …

Taxonomies of human activities

- Activities of Daily Living (ADLs) – elder care
- Compendium of Physical Activities – health/fitness
- OpenCyc – knowledge base
  - http://www.cyc.com/cyc/technology/whatiscyc_dir/whatdoescycknow
- And many more . . .

- None is complete
- None is explicit about overlap
- None discusses compound activities

Narrow vs. broad domains

- **May be hopeless to define a complete taxonomy**
- **Target domain can narrow scope**
  - e.g., physical exertion for fitness
  - e.g., ADLs for elder care

Labelling the data streams

- Each window requires a label
- Collect negative and positive examples of each activity
- Is an activity with one label a negative example for all others?
- What about unlabelled areas?

Classification

- How do we build a classifier for our feature streams?
- What features do we use? in what order?
- How do we train the classifier?
- How do we reconcile the output of different classifiers?
- How computationally expensive is classification?
- Can training be continued on-line
  or is it strictly an off-line process?

Training vs. Testing

- **Training**
  - process of learning how to classify
  - use lots of positive and negative examples
  - do examples generalize well?
    - do we have the right features we can use to tell them apart?
    - do they cover enough of the range?
  - can be computationally expensive
  - often off-line process
- **Testing**
  - process of classification given a new example
  - must be computationally efficient
  - often real-time on-line requirement

Thresholding

- **Collect all negative examples**
- **Collect all positive examples**
- **Take average of two mean values**

- **Which features do we use?**
  - have to yield a single Boolean or scalar value
  - easy to compute mean
- **Which feature do we start with?**

Decision tree (a simple approach)

- **Start with one feature test at the root**
  - partitions examples into two groups
  - if all within a group are correct – "pure" group – can stop – leaf node
  - if not all correct, do another test to further partition
- **Decide on the next feature that best partitions this group into "pure" groups**
- **Repeat**
  - until all groups are "pure" – in the limit, one example per leaf node
  - decide on when error is acceptable and stop

Voting

- **Make separate decisions based on different features**
- **Use very simple classifiers – one feature**
  - a “decision stump” – one level decision tree
  - e.g., if (vehicle weight > X kgs) then (it is a truck)
- **Count how many decisions classify into one category and go with maximum**

Simple algorithm

- **Pick a feature**
  - use training examples to find threshold
  - selection based on features that has least errors on training set
- **Pick another feature and another and another**
  - selection based on minimizing number of misclassifications from voting of features selected so far
  - weigh each feature's vote by how well it does

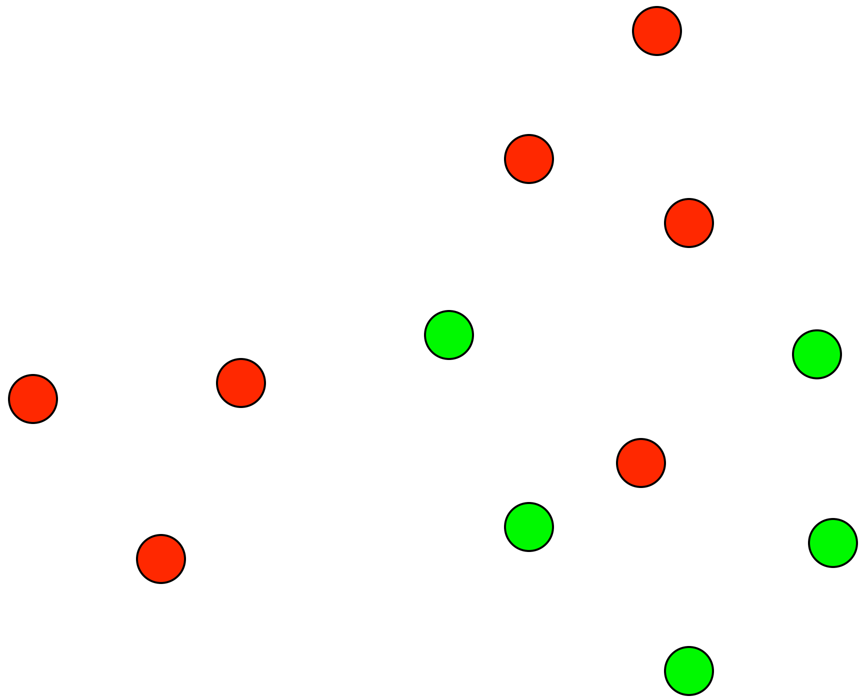
- **Not very good in practice**

Boosting (following slides derived from Phil Long at Columbia)

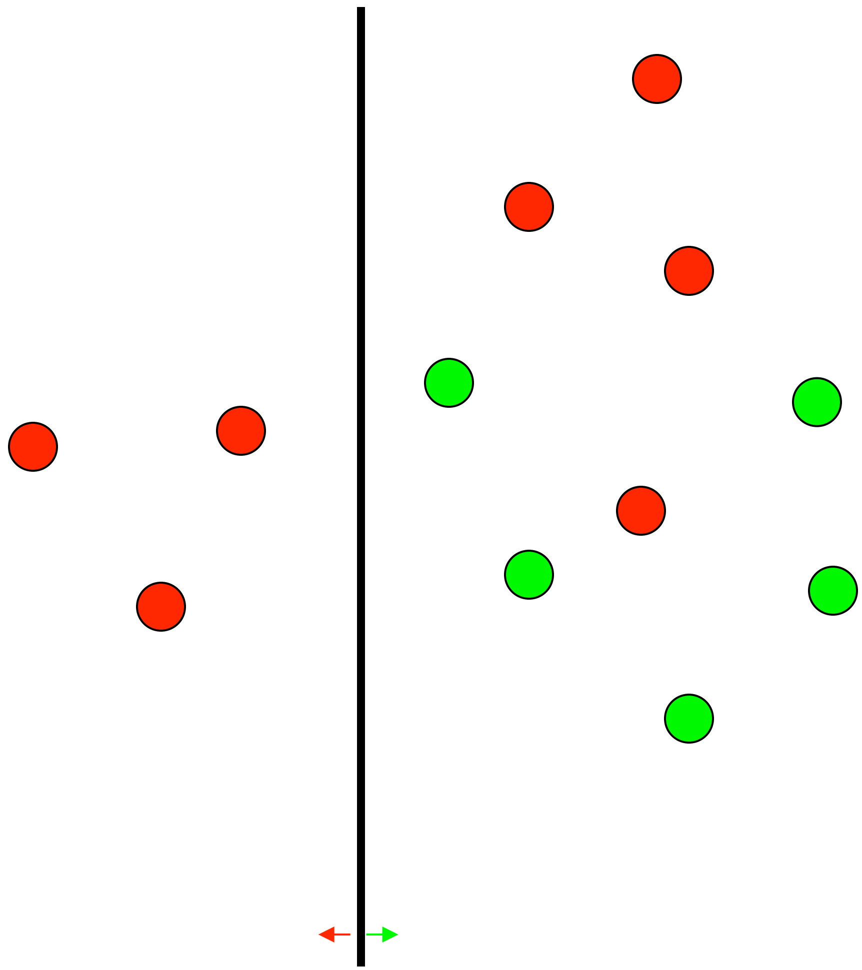
- **After selecting each feature**
  - reweight examples
  - more weight where previously chosen stumps were wrong
  - less weight where previously chosen stumps were right
  - emphasizes errors so next feature will be chosen to help with these
  - choose stump that minimizes weighted training error

- **Big practical success**

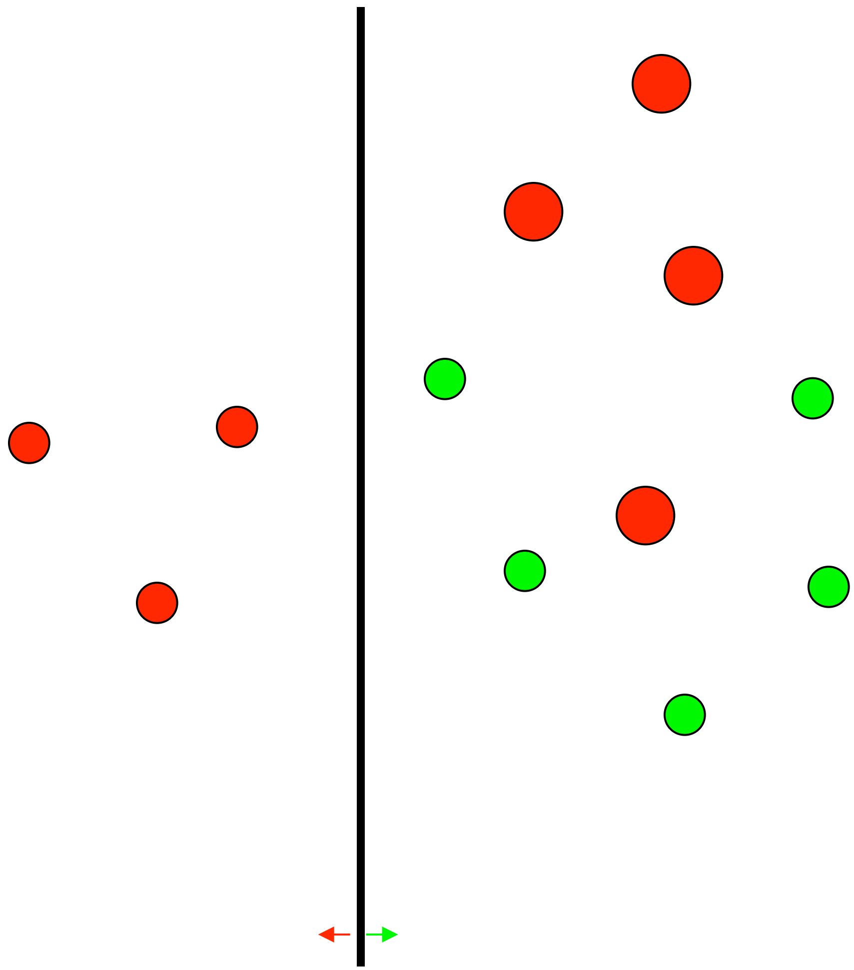
A simple example



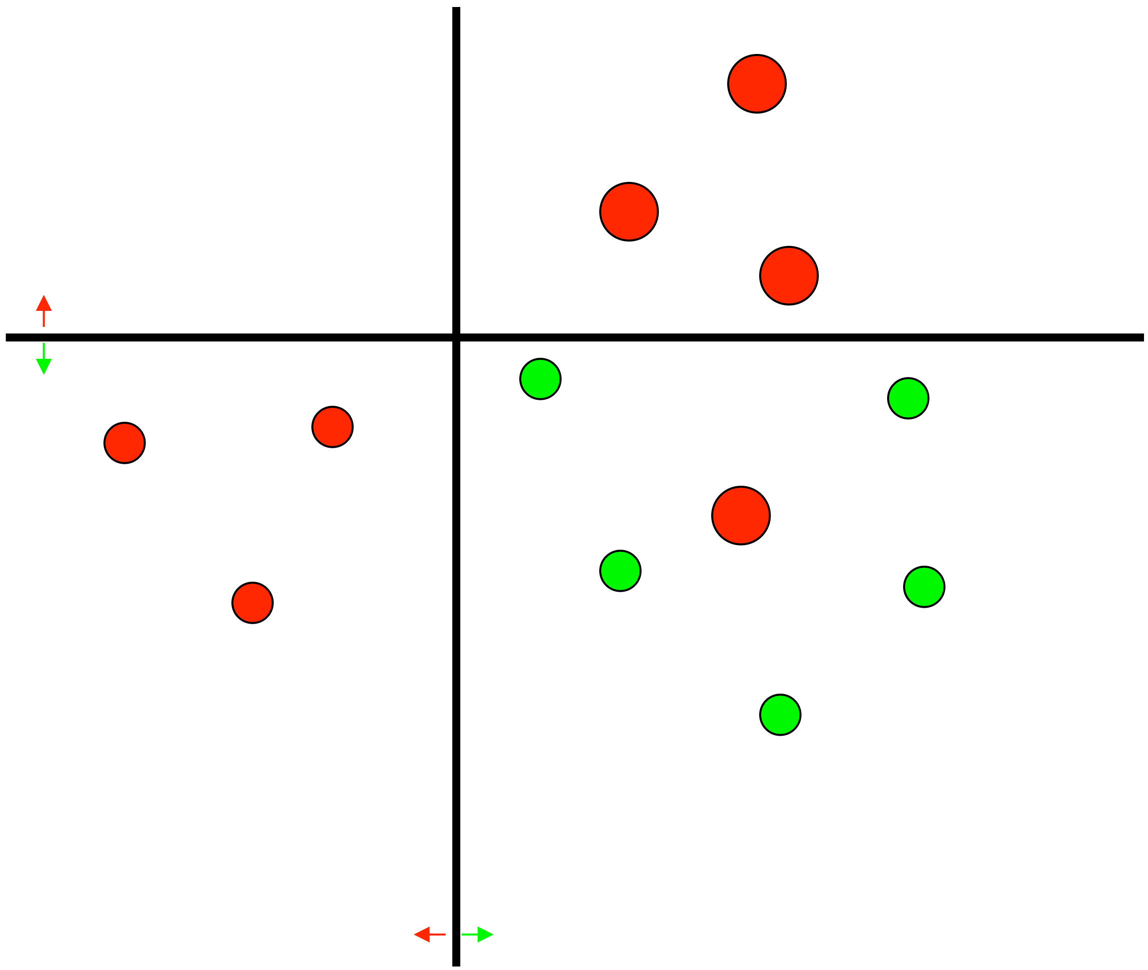
A simple example



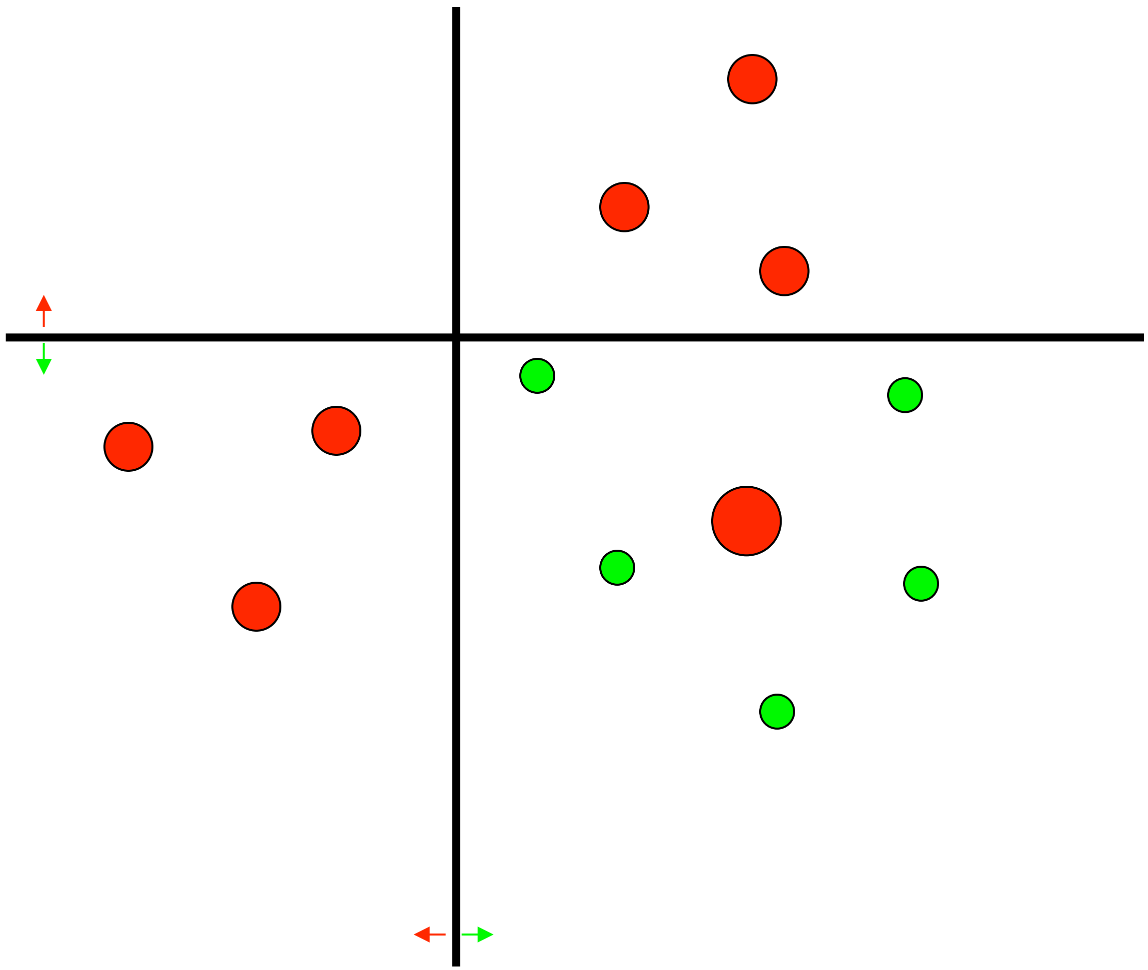
A simple example



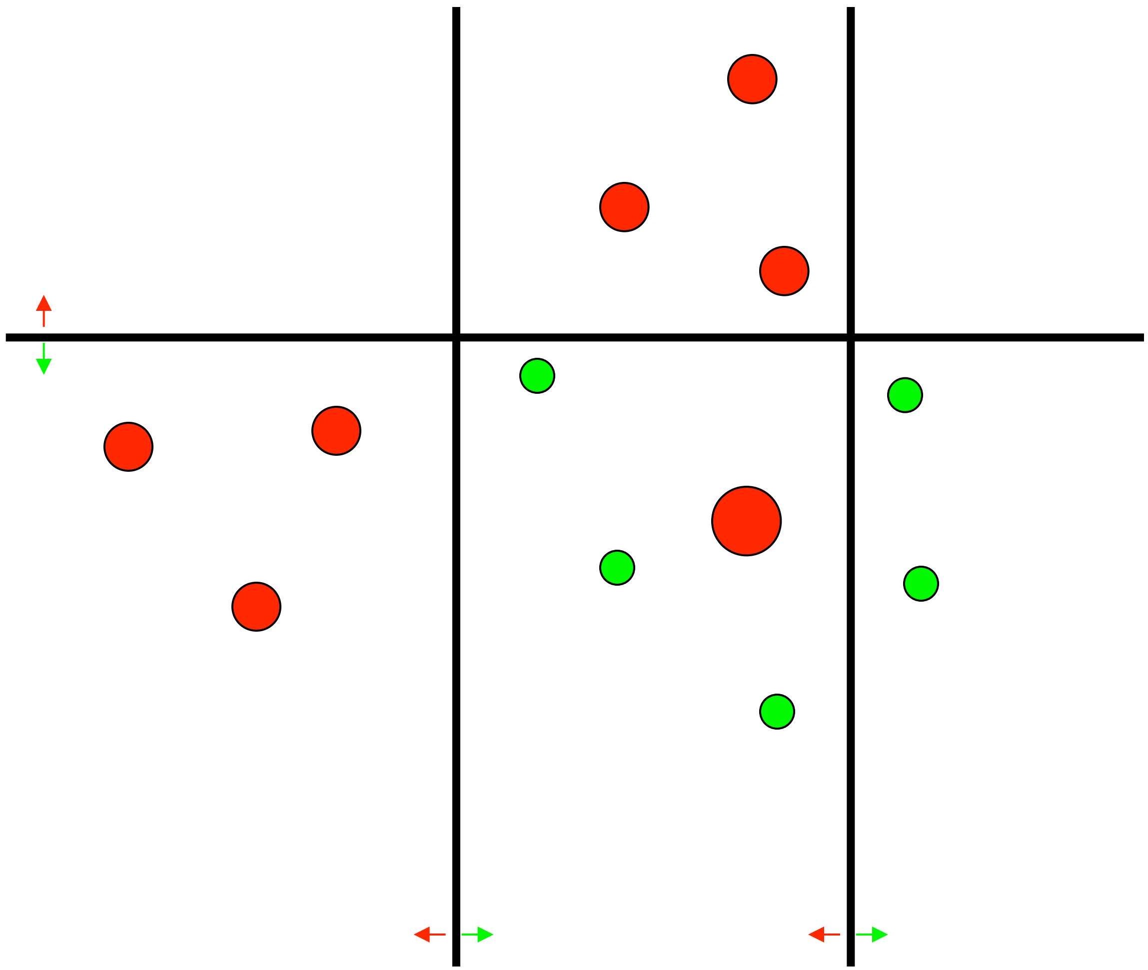
A simple example



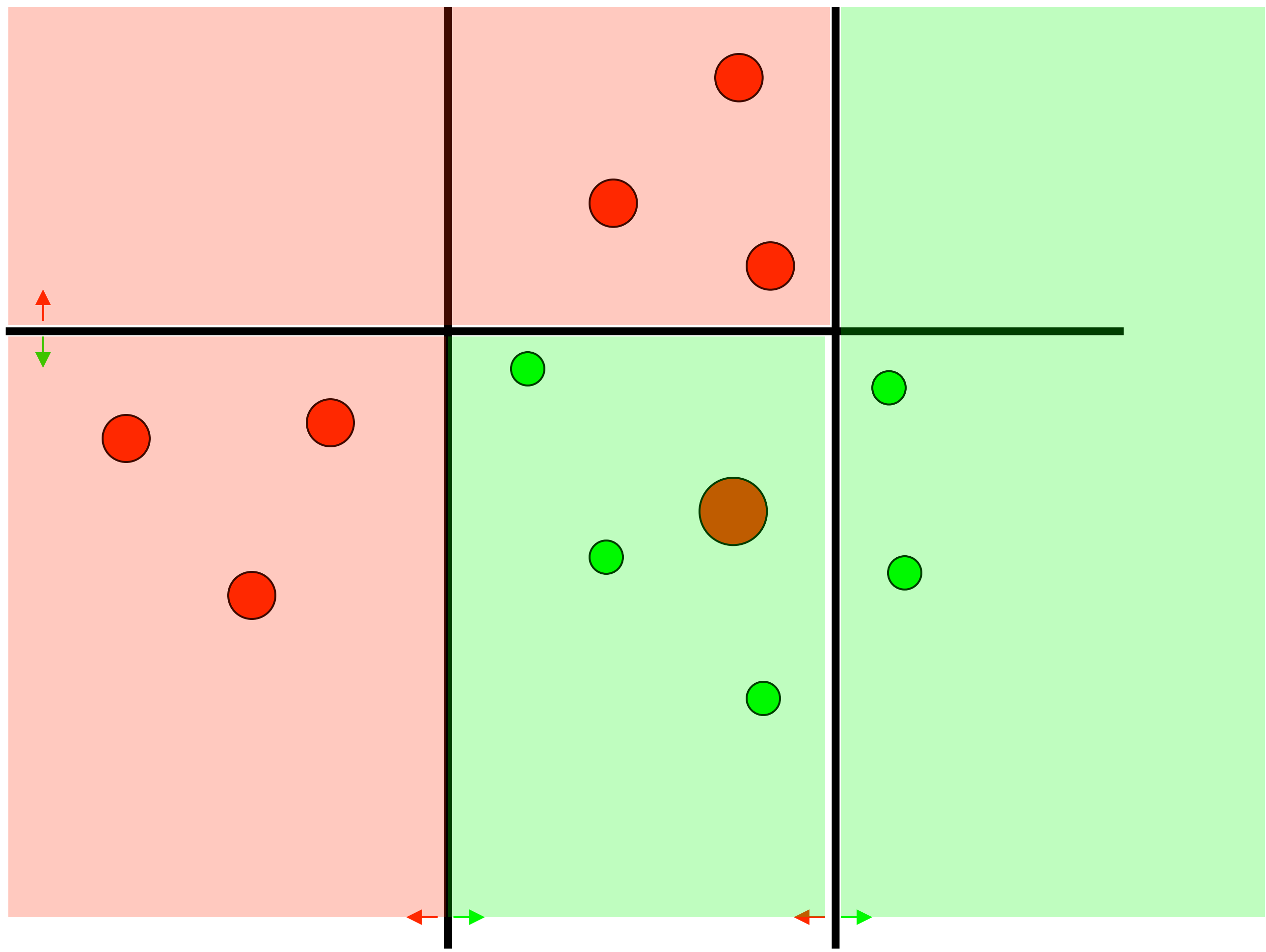
A simple example



A simple example



A simple example



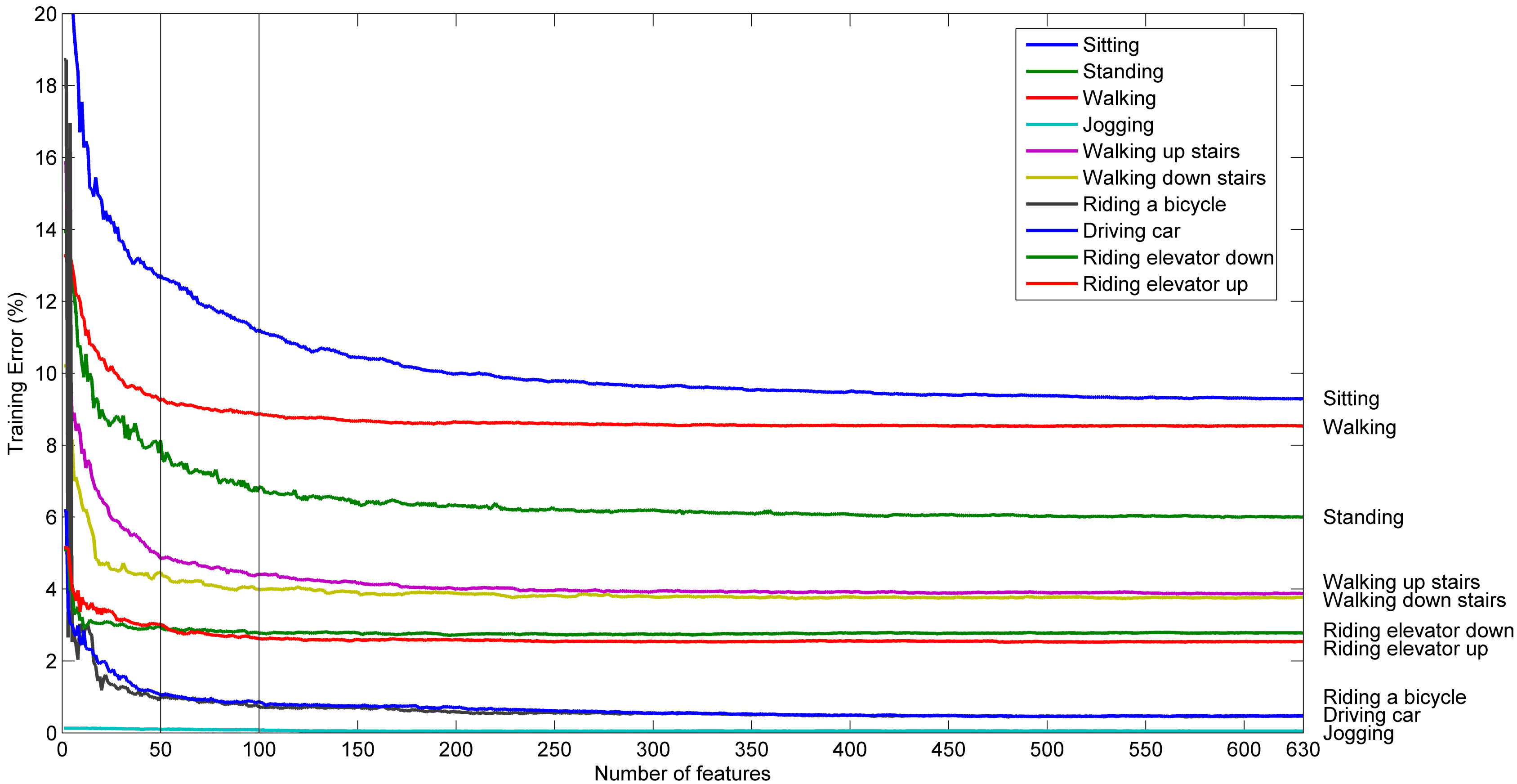
Boosting for features selection and classification

For a set of activities and a set features extracted from the sensors

Iteratively find features:

1. Select the feature that minimizes error for a chosen type of classifier ε_i
2. Calculate the classification error for that feature
3. Re-weight the training data so that the misclassified data gets more weight – the weight is a function of the error
4. Repeat steps 1-3

How many features are necessary?



For each activity A^i,select a threshold τ^i for the number of features such that the improvement obtained by adding more features is minimal

$$\Delta(\mathrm{error}(C^i(f^i_{r_1},\ldots,f^i_{r_N})),\mathrm{error}(C^i(f^i_{r_1},\ldots,f^i_{r_\tau}))) \le \varepsilon$$

Which are the most useful sensors?

	Audio	Accelerometer	Hi-Freq Vis Light	Digital Compass	Visible Light	IR Light	Ambient Light (IR-Vis)	Barometric Pressure	Temperature from Bar	Relative Humidity	Temp. from Relative Humidity
Brushing Teeth	40.0%	24.0%	6.0%		2.0%	4.0%	6.0%	6.0%	4.0%	6.0%	2.0%
Driving a car	24.0%	42.0%			4.0%	2.0%	12.0%	12.0%		2.0%	2.0%
Eating	24.0%	44.0%		2.0%	4.0%	2.0%	2.0%	12.0%	2.0%	4.0%	4.0%
Jogging	8.0%	50.0%	10.0%	2.0%		4.0%	4.0%	8.0%	6.0%	4.0%	4.0%
Riding elevator down	28.0%	30.0%	4.0%	4.0%		4.0%		16.0%	4.0%	4.0%	6.0%
Riding elevator up	18.0%	40.0%	2.0%	4.0%				22.0%	4.0%	4.0%	6.0%
Scrubbing Dishes	28.0%	32.0%	4.0%	2.0%		2.0%	4.0%	12.0%	2.0%	10.0%	4.0%
Sitting	28.0%	32.0%	2.0%	4.0%		4.0%	6.0%	14.0%	4.0%	4.0%	2.0%
Standing	28.0%	26.0%	4.0%	2.0%	2.0%	6.0%		10.0%	8.0%	10.0%	4.0%
Vacuuming	40.0%	38.0%	2.0%	4.0%		2.0%		12.0%			2.0%
Walking	24.0%	44.0%		4.0%	2.0%	2.0%	4.0%	10.0%	2.0%	6.0%	2.0%
Walking down stairs	28.0%	28.0%		6.0%		8.0%	2.0%	20.0%	2.0%	4.0%	2.0%
Walking up stairs	24.0%	46.0%	2.0%	6.0%	2.0%			16.0%	2.0%	2.0%	
Watching TV	2.0%	38.0%	2.0%	4.0%	2.0%	12.0%		4.0%	4.0%	22.0%	10.0%
Working on Computer	58.0%	26.0%				2.0%	2.0%	10.0%		2.0%	

Accuracy, precision, and recall

- **Important measures for classifiers**
- **Accuracy = overall correct classifications**
  - A = true-positives + true-negatives / all
- **Precision = percentage of positive classifications that are true**
  - P = items correctly classified / all items classified the same way
  - P = true-positives / true-positives + false-positives
- **Recall = percentage of positive classifications over all that should have been classified as positive**
  - R = items correctly classified / all items that should have been
  - R = true-positives / true-positives + false-negatives

Meaning of accuracy, precision, and recall

- Accuracy = 1.0 means all classifications were correct
- Precision = 1.0 means that every item classified as an X is indeed an X (but says nothing about the number of items that were also Xs that were not classified correctly)
- Recall = 1.0 means that every X was classified as being an X (but says nothing about how many other items were incorrectly classified as being Xs)

Accuracy of the static classifiers: dataset 1

		Classified Activity (by Decision Stumps)									
		Sitting	Standing	Walking	Jogging	Walking up stairs	Walking down stairs	Riding a bicycle	Driving car	Riding elevator down	Riding elevator up
Precision	Labeled Activities: Sitting	90.9%	43.3%	1.1%	0.3%	2.6%	2.7%	7.2%	10.2%	9.0%	5.6%
	Standing	7.1%	44.9%	0.3%		0.9%	0.3%	1.8%	0.7%	1.5%	1.5%
	Walking	1.2%	8.7%	95.1%	1.3%	21.1%	12.9%	5.4%	1.3%	0.8%	0.7%
	Jogging	0.0%		0.1%	98.3%		0.1%				
	Walking up stairs	0.0%	0.1%	1.9%		73.6%	0.7%	0.1%	0.0%		0.2%
	Walking down stairs		0.0%	1.4%	0.1%	1.0%	83.0%	0.1%		0.5%	
	Riding a bicycle	0.1%	0.1%	0.2%				85.3%			
	Driving car	0.5%	0.0%	0.0%		0.2%		0.1%	87.7%	0.1%	0.2%
	Riding elevator down	0.1%	1.7%			0.1%	0.1%		0.0%	87.5%	0.4%
	Riding elevator up	0.1%	1.2%	0.0%		0.5%			0.1%	0.5%	91.4%

		Sitting	Standing	Walking	Jogging	Walking up stairs	Walking down stairs	Riding a bicycle	Driving car	Riding elevator down	Riding elevator up
Recall	Labeled Activities: Sitting	86.6%	10.0%	0.8%	0.0%	0.1%	0.1%	0.8%	1.3%	0.2%	0.1%
	Standing	38.2%	58.2%	1.3%		0.3%	0.1%	1.0%	0.5%	0.2%	0.2%
	Walking	1.6%	2.7%	92.6%	0.0%	1.5%	0.7%	0.7%	0.2%	0.0%	0.0%
	Jogging	0.1%		1.8%	97.7%		0.3%				
	Walking up stairs	0.1%	0.2%	26.1%		72.8%	0.6%	0.1%	0.1%		0.1%
	Walking down stairs		0.1%	22.7%	0.1%	1.2%	75.4%	0.3%		0.3%	
	Riding a bicycle	0.8%	0.3%	1.3%				97.6%			
	Driving car	4.1%	0.0%	0.1%		0.1%		0.1%	95.6%	0.0%	0.1%
	Riding elevator down	3.4%	14.4%			0.2%	0.2%		0.1%	81.3%	0.3%
	Riding elevator up	3.2%	10.0%	0.1%		1.0%			0.4%	0.4%	84.8%

Tested on over 12 hours of wearable-sensor data collected by two volunteers

Accuracy of the static classifiers: dataset 2

Right Shoulder — Classified Activity (by Static Classifier)

Precision — Labeled Activities

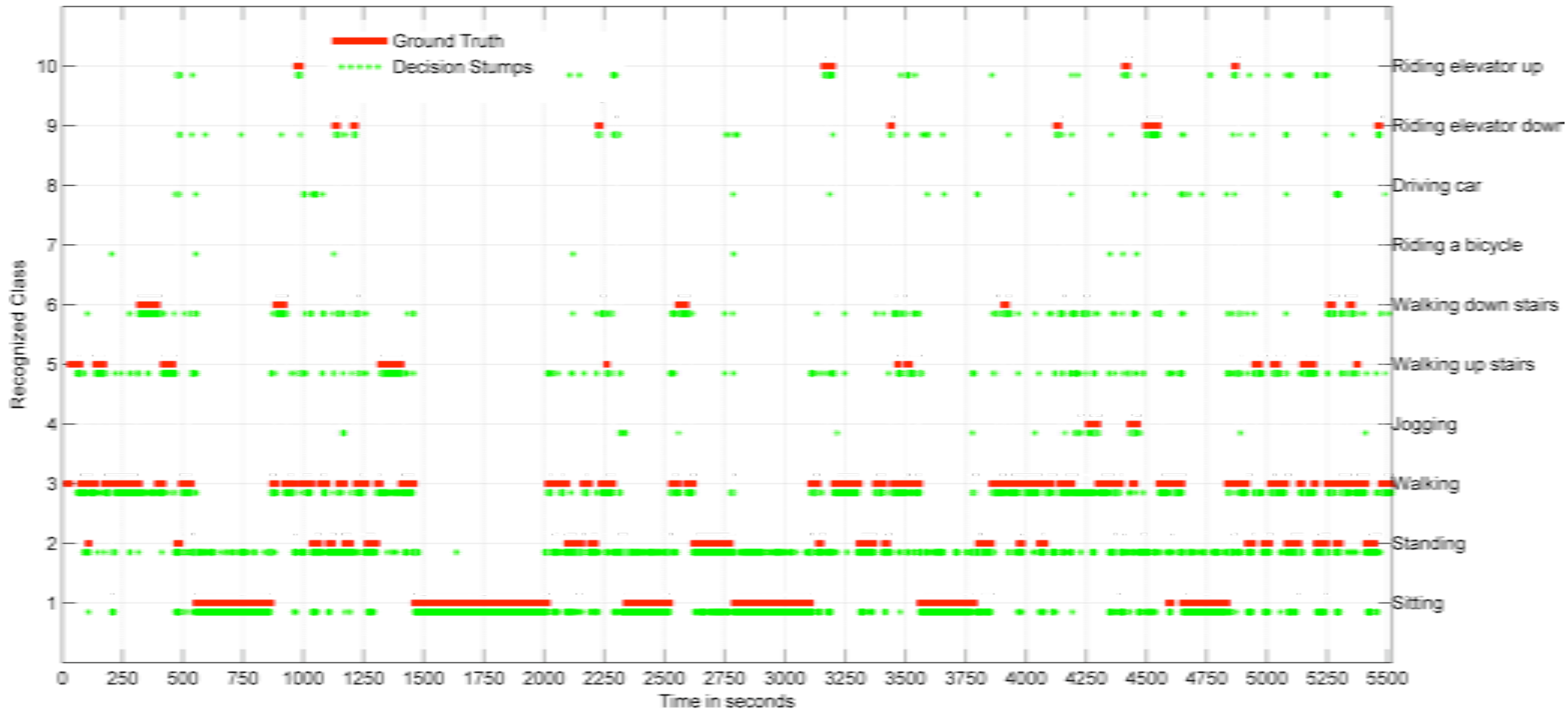
Labeled Activities	Brushing Teeth	Driving a car	Eating	Jogging	Riding elevator down	Riding elevator up	Scrubbing Dishes	Sitting	Standing	Vacuum-ing	Walking	Walking down stairs	Walking up stairs	Watching TV	Working on Computer
Brushing Teeth	**63.6%**		1.5%		0.3%	0.5%	10.1%		3.3%		1.3%		0.4%		
Driving a car	0.7%	**98.2%**	0.5%		1.0%			1.6%	0.1%				0.9%	0.7%	
Eating	1.1%	0.0%	**78.7%**			0.3%	1.1%		0.9%		0.1%				3.8%
Jogging				**98.9%**					0.2%		0.2%		0.4%		
Riding elevator down	0.7%	0.3%	2.0%		**80.2%**	3.6%	1.1%	0.1%	5.4%		0.6%		1.3%	0.1%	0.4%
Riding elevator up	0.7%	0.2%	1.3%		3.0%	**84.4%**	0.8%		3.7%		0.8%		0.4%		0.9%
Scrubbing Dishes	18.0%	0.0%	2.6%		0.3%	0.5%	**72.8%**	0.2%	4.6%		0.3%	0.7%	0.4%		0.2%
Sitting	0.2%	0.4%	1.5%					**97.0%**	0.4%				1.3%	0.7%	
Standing	9.5%	0.0%	5.0%		8.1%	2.8%	10.4%	0.3%	**72.9%**		5.5%	1.4%	4.3%	0.1%	2.4%
Vacuuming						0.3%				**100.0%**					
Walking	5.0%	0.7%	2.3%	1.1%	4.8%	3.6%	3.5%	0.8%	7.0%		**84.9%**	11.0%	7.4%	0.1%	
Walking down stairs	0.2%		0.2%		1.8%				0.3%		3.5%	**84.8%**	1.3%		
Walking up stairs	0.2%	0.0%	0.5%		0.5%	3.8%	0.3%		1.0%		2.7%	2.1%	**81.8%**		
Watching TV		0.1%						0.1%						**98.1%**	
Working on Computer			3.6%			0.3%			0.2%						**92.3%**

Recall — Labeled Activities

Labeled Activities	Brushing Teeth	Driving a car	Eating	Jogging	Riding elevator down	Riding elevator up	Scrubbing Dishes	Sitting	Standing	Vacuum-ing	Walking	Walking down stairs	Walking up stairs	Watching TV	Working on Computer
Brushing Teeth	**68.1%**		3.4%		0.2%	0.5%	9.2%		14.4%		3.9%		0.2%		
Driving a car	0.1%	**98.5%**	0.2%		0.2%			0.7%	0.0%				0.1%	0.2%	
Eating	0.6%	0.1%	**92.4%**			0.1%	0.5%		2.1%		0.1%				4.0%
Jogging				**91.8%**					4.1%		3.1%		1.0%		
Riding elevator down	0.6%	1.5%	3.8%		**66.8%**	3.0%	0.8%	0.2%	20.3%		1.5%		0.6%	0.2%	0.6%
Riding elevator up	0.7%	0.9%	2.7%		2.7%	**73.5%**	0.7%		14.9%		2.2%		0.2%		1.6%
Scrubbing Dishes	16.7%	0.2%	5.1%		0.2%	0.4%	**57.8%**	0.4%	17.4%		0.8%	0.2%	0.2%		0.4%
Sitting	0.1%	0.9%	1.2%					**96.5%**	0.6%				0.3%	0.4%	
Standing	2.7%	0.1%	2.9%		2.0%	0.7%	2.5%	0.3%	**82.6%**		4.2%	0.1%	0.6%	0.1%	1.3%
Vacuuming						0.5%				**99.5%**					
Walking	1.7%	1.4%	1.6%	0.1%	1.4%	1.1%	1.0%	0.7%	9.6%		**78.9%**	1.2%	1.3%	0.1%	
Walking down stairs	0.5%		1.1%		3.8%				3.2%		23.2%	**66.5%**	1.6%		
Walking up stairs	0.4%	0.4%	1.9%		0.7%	5.6%	0.4%		6.7%		12.3%	1.1%	**70.5%**		
Watching TV		0.3%						0.2%						**99.5%**	
Working on Computer			4.2%			0.1%			0.4%						**95.3%**

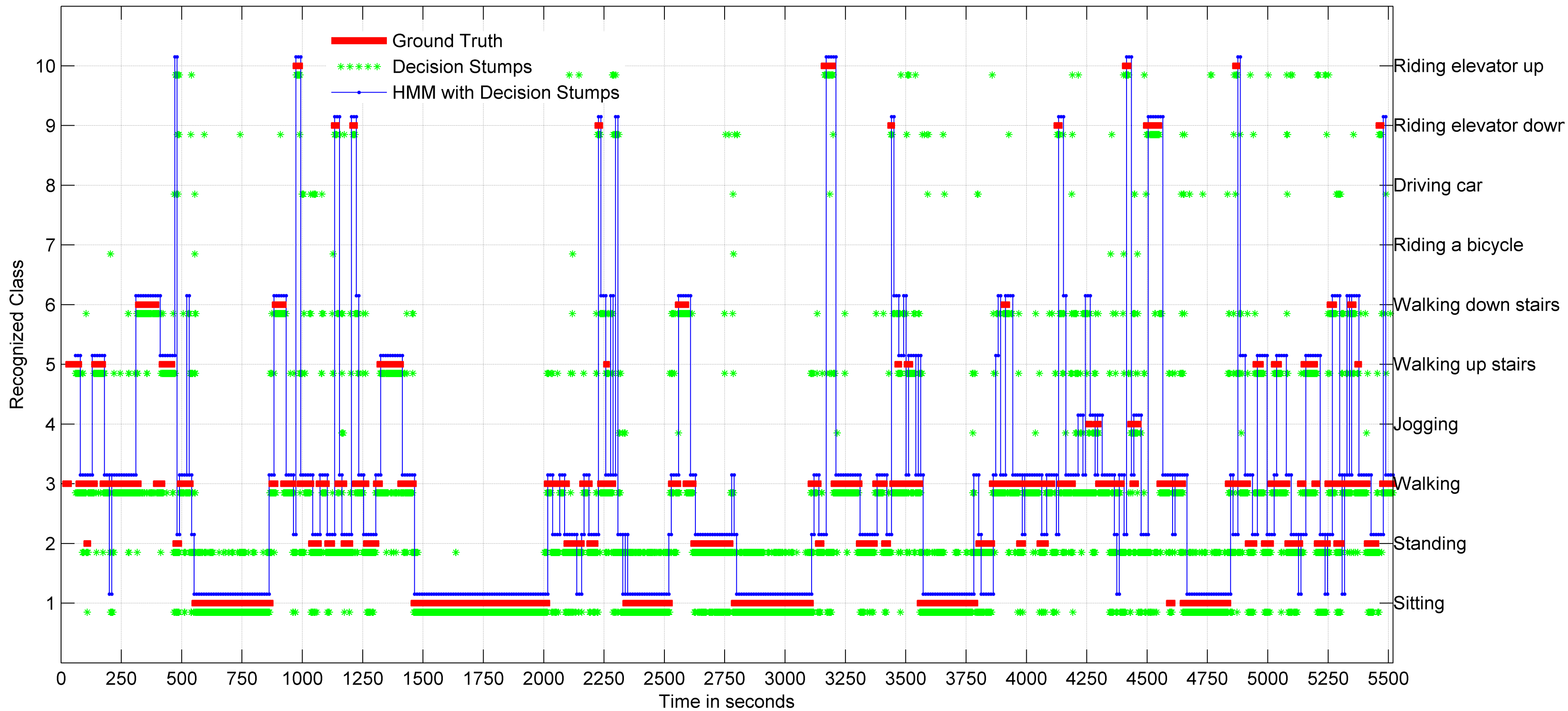
Tested on approximately an hour of wearable-sensor data collected by five volunteers

Example classification trace



Output of the decision stumps classifier (at 4Hz) in green and the ground truth in red for a continuous hour and half segment of data.

Looking at the classification trace again



Output of the decision stumps classifier in green (at 4Hz), HMM in blue with probabilities as inputs (using a 15 second sliding window with 5 second overlap), and the ground truth in red for a continuous hour and half segment of data.

Accuracy of the HMM classifiers: dataset 1

Precision

Labeled Activities	Classified Activity (by HMM)									
	Sitting	Standing	Walking	Jogging	Walking up stairs	Walking down stairs	Riding a bicycle	Driving car	Riding elevator down	Riding elevator up
Sitting	89.8%	38.5%	0.5%				0.4%	33.4%		
Standing	10.1%	50.8%	1.4%							
Walking	0.1%	7.4%	97.7%		5.2%	2.5%				
Jogging				100.0%						
Walking up stairs					94.8%					
Walking down stairs			0.5%			97.5%				
Riding a bicycle		3.3%					99.6%			
Driving car								66.6%		
Riding elevator down									100.0%	
Riding elevator up										100.0%

Recall

Labeled Activities	Sitting	Standing	Walking	Jogging	Walking up stairs	Walking down stairs	Riding a bicycle	Driving car	Riding elevator down	Riding elevator up
Sitting	87.5%	3.7%	0.1%				0.1%	8.6%		
Standing	65.6%	32.8%	1.6%							
Walking	0.4%	4.0%	93.8%		1.3%	0.4%				
Jogging				100.0%						
Walking up stairs					100.0%					
Walking down stairs			2.5%			97.5%				
Riding a bicycle		1.7%					98.3%			
Driving car								100.0%		
Riding elevator down									100.0%	
Riding elevator up										100.0%

Accuracy of the HMM classifiers: dataset 2

Precision

Rows: Labeled Activities; columns: recognized activity

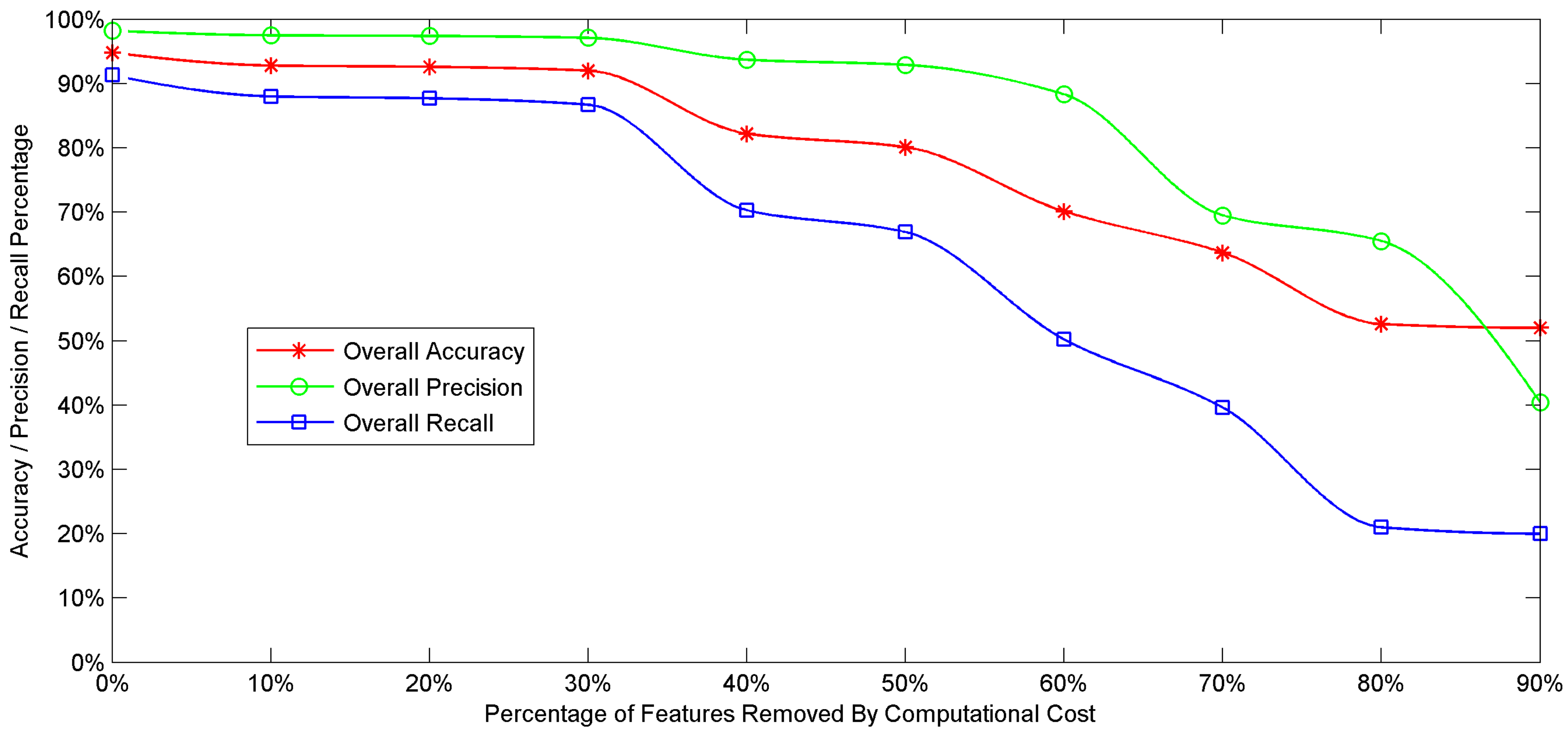
Labeled Activities	Brushing Teeth	Driving a car	Eating	Jogging	Riding elevator down	Riding elevator up	Scrubbing Dishes	Sitting	Standing	Vacuuming	Walking	Walking down stairs	Walking up stairs	Watching TV	Working on Computer
Brushing Teeth	95.83%														
Driving a car		98.68%				3.23%			1.06%		1.30%				
Eating			100.00%												
Jogging				100.00%											
Riding elevator down					93.75%										
Riding elevator up						87.10%									
Scrubbing Dishes							100.00%		1.06%						
Sitting	4.17%							85.42%							
Standing		1.32%			6.25%	9.68%		14.58%	97.87%						
Vacuuming										100.00%					
Walking											98.70%				
Walking down stairs												100.00%			
Walking up stairs													100.00%		
Watching TV														100.00%	
Working on Computer															100.00%
	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

Recall

Rows: Labeled Activities; columns: recognized activity

Labeled Activities	Brushing Teeth	Driving a car	Eating	Jogging	Riding elevator down	Riding elevator up	Scrubbing Dishes	Sitting	Standing	Vacuuming	Walking	Walking down stairs	Walking up stairs	Watching TV	Working on Computer
Brushing Teeth	100.00%														
Driving a car		98.04%				0.65%			0.65%		0.65%				
Eating			100.00%												
Jogging				100.00%											
Riding elevator down					100.00%										
Riding elevator up						100.00%									
Scrubbing Dishes							93.75%		6.25%						
Sitting	1.20%							98.80%							
Standing		1.77%			1.77%	2.65%		12.39%	81.42%						
Vacuuming										100.00%					
Walking											100.00%				
Walking down stairs												100.00%			
Walking up stairs													100.00%		
Watching TV														100.00%	
Working on Computer															100.00%

Computational cost



As the number of available features are removed, based upon the computational cost, the overall accuracy, precision, and recall gradually decline.

Effect of Sensor Placement

The system can classify activities, equally well (within ±1%), on several parts of the body where consumer devices are normally carried: waist, shoulder (e.g. a backpack strap), and wrist (e.g. a wrist-watch)

	Data Set 1	Data Set 2		
	Shoulder	Shoulder	Waist	Wrist
Overall Static Classifier Accuracy	91.3%	90.7%	92.0%	91.5%
Overall Static + HMM Classifier Accuracy	95.0%	99.0%	98.1%	99.4%

J. Lester, T. Choudhury, N. Kern, G. Borriello and B. Hannaford. *A Hybrid Discriminative/Generative Approach to Modeling Human Activities*. To appear in the proceedings of IJCAI 2005

Next steps: higher-level behaviors

- **Combine sensor data with location data obtained from WiFi, GPS, GSM etc.**
  - use activities to categorize locations
  - use location to narrow choices of activities
- **Model varying time granularities**
- **Learn transition patterns between activities**
  - these can be complicated by inter-leaved activities
  - activities that may not have clear starts and stops
- **Goal: interpret a day in a person's life with enough detail to provide assistive services, automatic journaling, etc.**

Summary

- **Personal sensing devices for activity recognition**
  - cell phone is the likely form-factor
- **Context-aware applications**
  - using activity and location information
- **High precision and recall for short-time scale activities**
  - combining the current primitive activities to model more complex behaviors – e.g. activities of daily living
- **Capture social roles and relationships by sensing attributes of interactions**
  - mine social networks and communication records