SENSING SYSTEMS

CSE 599 N1: Modern Mobile Systems

modernmobile.cs.washington.edu

Content borrowed from Hrvoje Benko Jon Froehlich, Alex Mariakakis, Don Patterson, Shwetak Patel, Gregory Abowd

Plan

- Lecture
    - Belloti’s 5 questions
    - Gierad’s Viband
    - Example of a signal processing pipeline
    - Chris Harrison’s Omnitouch
- Discussion

Papers

Making Sense of Sensing Systems: Five Questions for Designers and Researchers (CHI 2002)

ViBand: High-Fidelity Bio-Acoustic Sensing Using Commodity Smartwatch Accelerometers (UIST 2016)

OmniTouch: Wearable Multitouch Interaction Everywhere (UIST 2011)

BELLOTI’S 5 QUESTIONS

Challenges with this vision?
How to design it well?

Transition in how we think about interfaces

Interaction as gulf of execution and evaluation (Norman)

“the gulf of execution is the difference between the intentions of the users and what the system allows them to do or how well the system supports those actions”

“A gap between the psychological language (or mental model) of the user's goals and the very physical action-object language of the controls of the VCR via which it is operated.”

Gulf of execution

User thinks: "Hit the record button,"

Correct sequence is

1) Hit the record button.
2) Specify time of recording via the controls X, Y, and Z.
3) Select channel via the channel-up-down control.
4) Press the OK button.

E.g. command lines and GUIs

Interaction as conversation

E.g. speech interfaces, ubicomp, human-robot interfaces, tangible interfaces, VR/AR

Why conversations are challenging

Potentially imprecise

Multimodal

Context is important

Not all options are visible/discoverable

Norman’s seven stages of action with system interaction

1. Forming the goal
2. Forming the intention
3. Specifying an action
4. Executing the action
5. Perceiving the state of the world
6. Interpreting the state of the world
7. Evaluating the outcome

Focuses on user cognition, that is the user’s mental model.

Belloti et al. Designing sensing systems

Goal: Systematic framework for the design of sensing systems

Focus here is on communicative aspects of interaction. Focus is on the joint accomplishments of the user and system.

Genres: A set of design conventions anticipating particular usage contexts with their own conventions

Systems genres: games, productivity tools, appliances

Interaction genres: GUI, voice activation, remote control

Belloti et al.
Five questions for designers of sensing systems

Address: directing communication to a system

Attention: establishing that the system is attending

Engagement

Action: defining what is to be done with the system

Alignment: monitoring system response

Feedback
Error recovery

Accident: avoiding or recovering from errors or misunderstandings

Belloti et al.

Each of the five basic questions can be easily solved with the GUI paradigm and analogue counterparts.

But when designing new sensing systems, each of the questions take center stage.

Bellotti et al.: Five questions

Address:

- How do I address the system
- How do I address one (or more) of the many possible devices?
- How not to address the system?

GUI: keyboard, mouse, physical access

Sensing systems: Deal with signal vs. noise, disambiguate intended target, *not* address the system. Deal with no/unwanted responses

- Magic keyboard (“Alexa”, “Xbox”, “Siri”)
- Magic pose (Xbox: hand in front of the body)

Bellotti et al.: Five questions

Attention:

- How do I know the system is ready and attending to my actions?

GUI: graphical feedback, conventions (blinking cursor)

Sensing systems:

- What is appropriate feedback for mid-air interaction?
- How about a conversational agent
- Direct user to a particular zone
- Problems: privacy/security, wasted input effort

Bellotti et al.: Five questions

Action:

- How do I affect a meaningful action?
- How to control its extent?
- How to specify a target or targets for my action?

GUI: click to select, click+drag to multiple select etc.

Sensing systems:

- Selection mechanism? How to “click”? How to “clutch”? Operations are more limited. Deal with unintended actions, or failure to execute actions.

Bellotti et al.: Five questions

Alignment:

- How do I know the system is doing (has done) the right thing?

GUI: feedback conventions, progress bars, drag+drop interactions, etc.

Sensing systems: how to make system state perceivable, or query-able?

Problems: inability to detect mistakes, unrecoverable state, difficulty evaluating new state, failure to execute action, inability to differentiate more than limited action space

Bellotti et al.: Five questions

Accident:

- How do I avoid mistakes? How to recover from them? Control/guide direct actions, stop/cancel/undo/delete

GUI: Direct manipulation, undo, delete, preview actions.

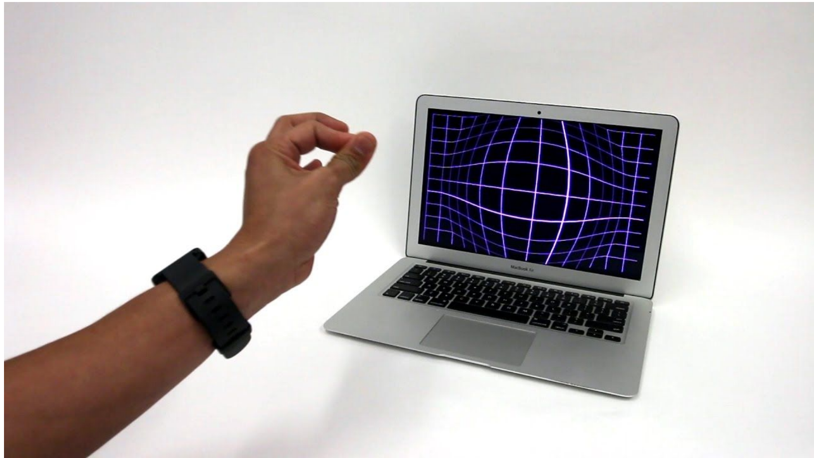
Sensing systems:

Unintended actions? How to undo? How to cancel action in progress?

VIBAND

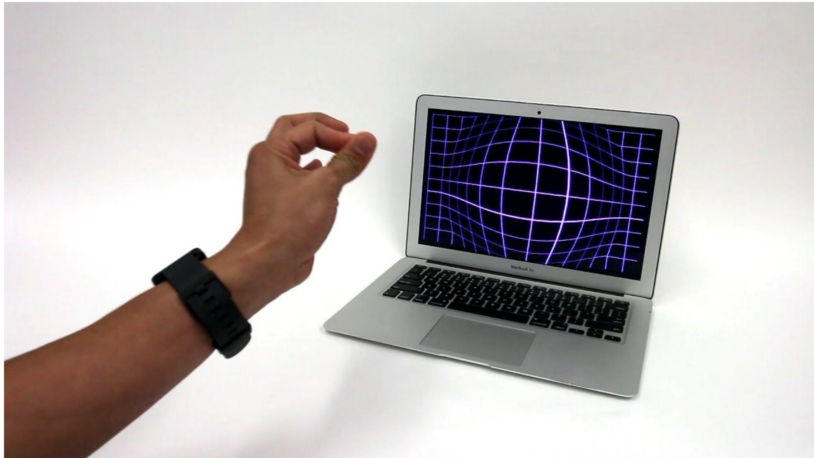
Viband

- Summarize the paper?



Viband

- Take advantage of unique context provided by smartwatches and wearables -- they reside on the body! -- always available input and interaction
- Captures bio-acoustic signals at 4kHz by overclocking the smartwatch's accelerometer
- Capture tiny micro-vibrations propagating along the arm

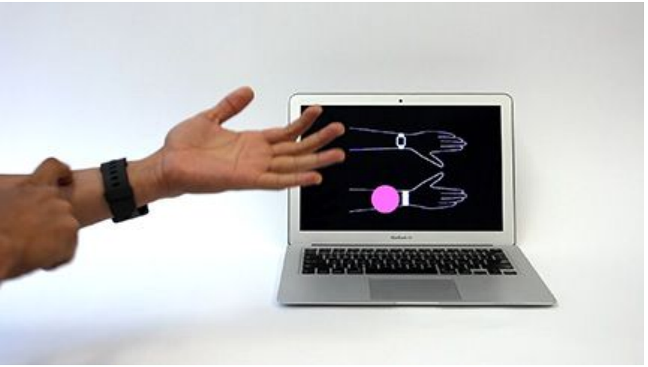


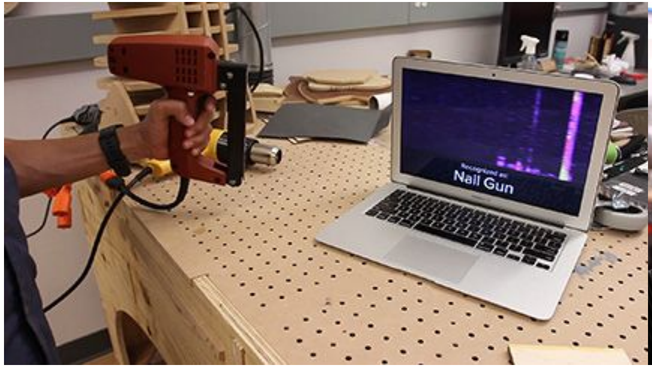
Applications

What are some of them?

Applications

1. Classify hand gestures+on-device motion tracking
2. Detect and classify vibrations of uninstrumented mechanical/motor objects
3. Structured vibrations for data communication through human body





Fast fourier transform

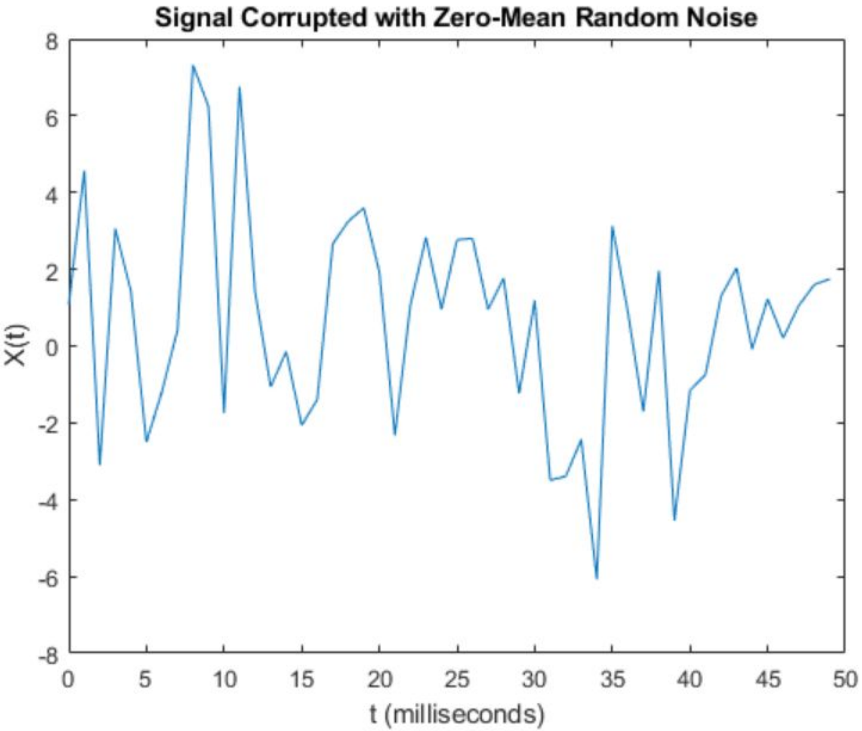
Has anyone heard of this?

What does it do?

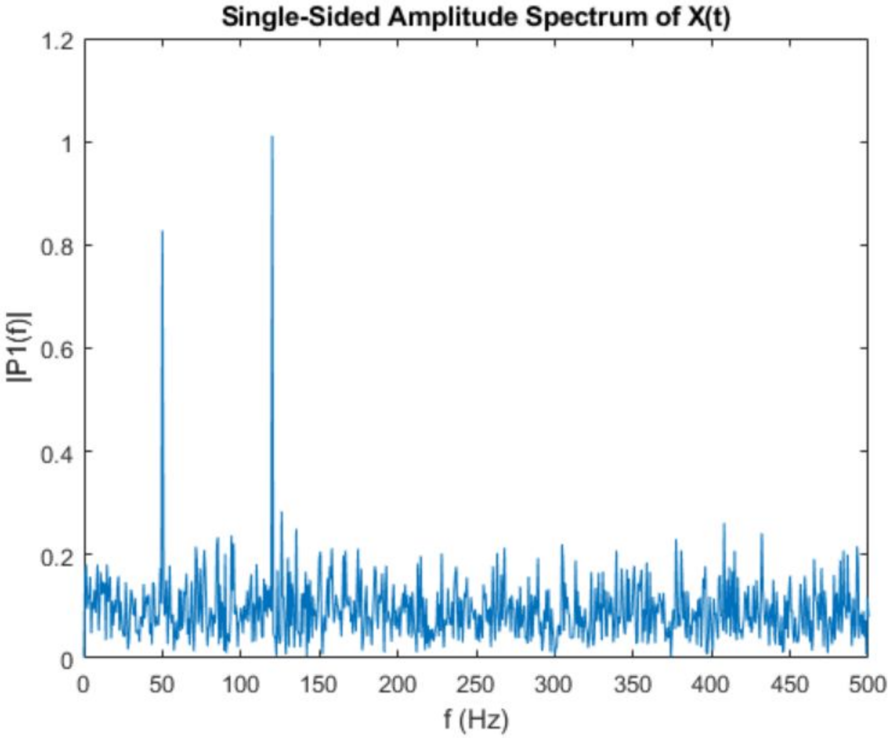
Why is it so important?

Fast fourier transform

```
S = 0.7*sin(2*pi*50*t) + sin(2*pi*120*t);
```



Signal Corrupted with Zero-Mean Random Noise



Single-Sided Amplitude Spectrum of X(t)

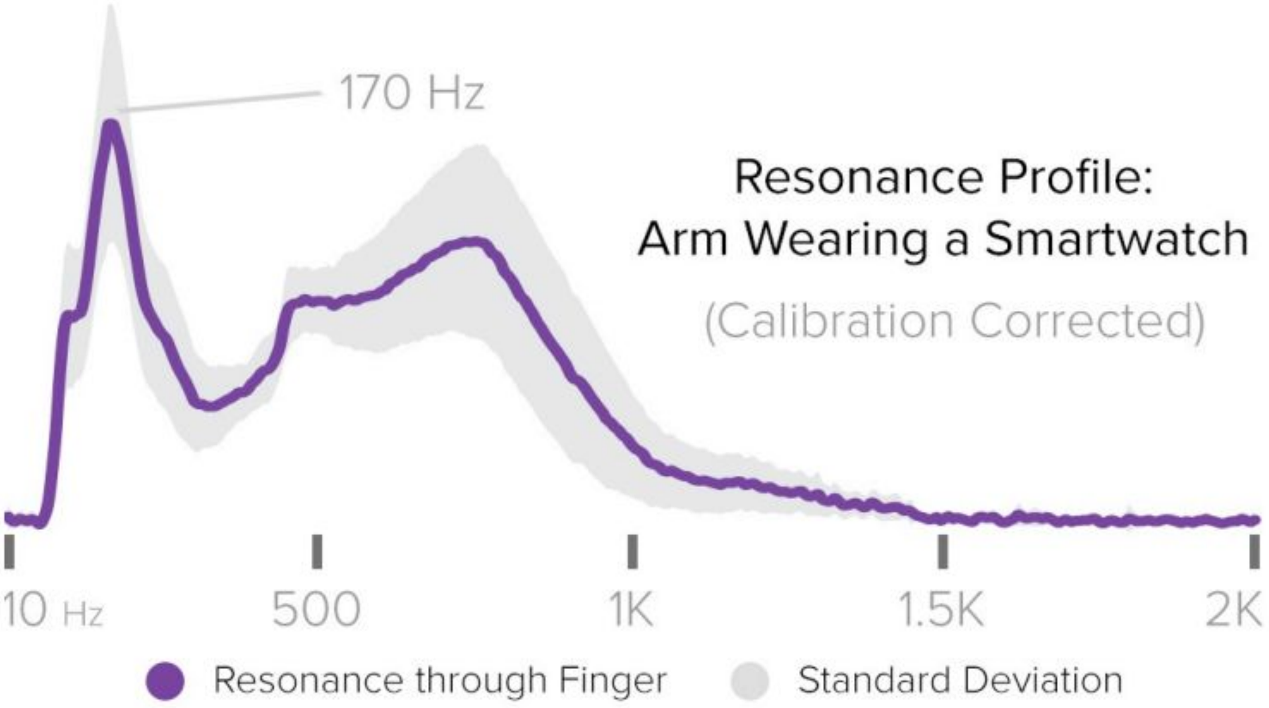
Resonance

What is it?

Why is it important?

How can it be used to guide our design of a sensing system?

Resonance



For human arm resonance is 20Hz - 1kHz
Salient peaks at 170Hz and 750Hz

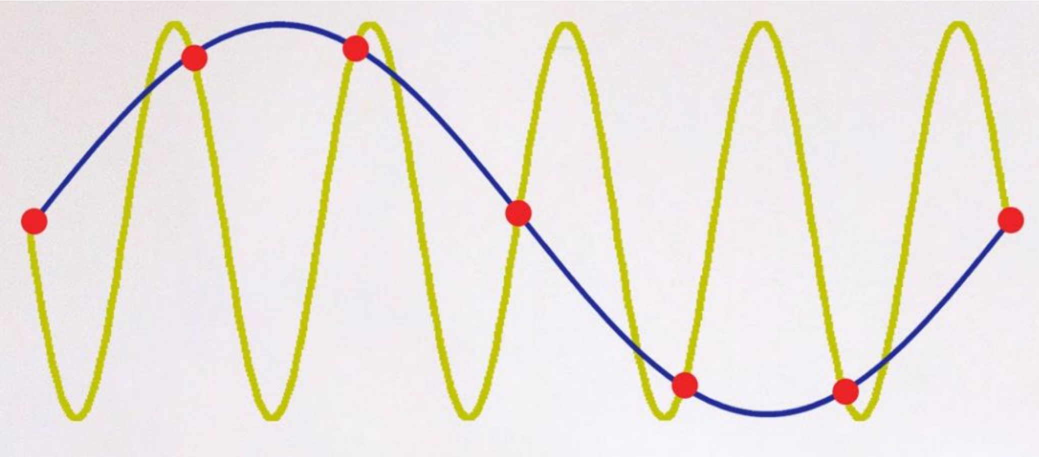
For ear drums it is 2-4kHz
For sinuses it is 1-2kHz

Nyquist theorem

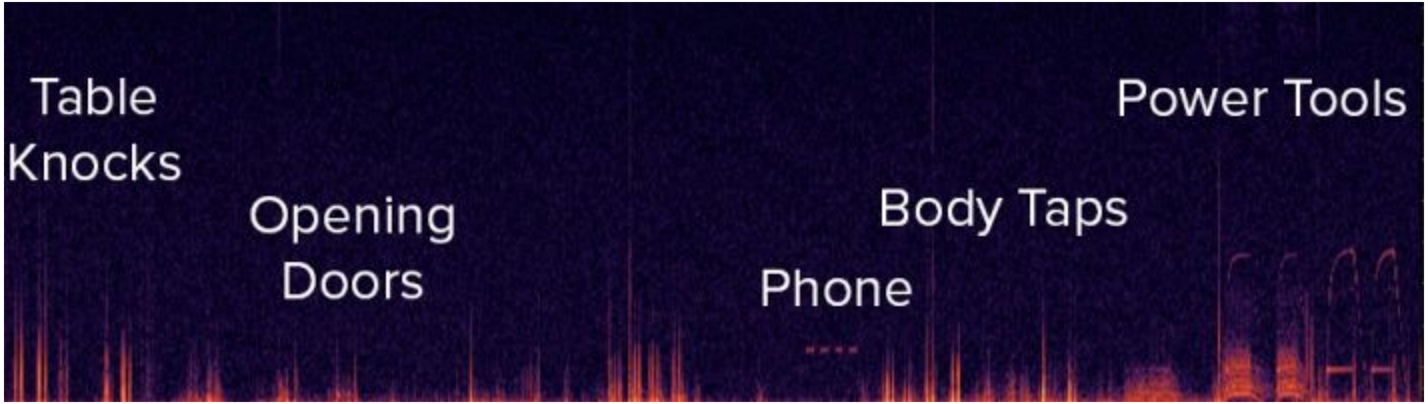
E.g. If you sample at 4kHz, you can only capture frequencies up to 2kHz

The higher the sampling rate the better the resolution

If too low you have aliasing



Spectrogram



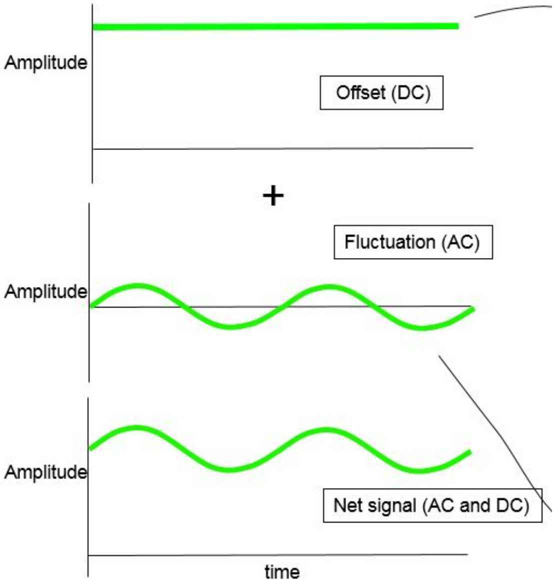
Frequency

Time

Color=intensity, often power in dB
Frequency/time axes may be reversed

Preprocessing of signals

- Remove DC component -- DC component is just the mean of the waveform
  Ensures waveform is centered at 0
- Moving average to smooth waveform
- Envelope
- Filters (highpass, lowpass, bandpass)



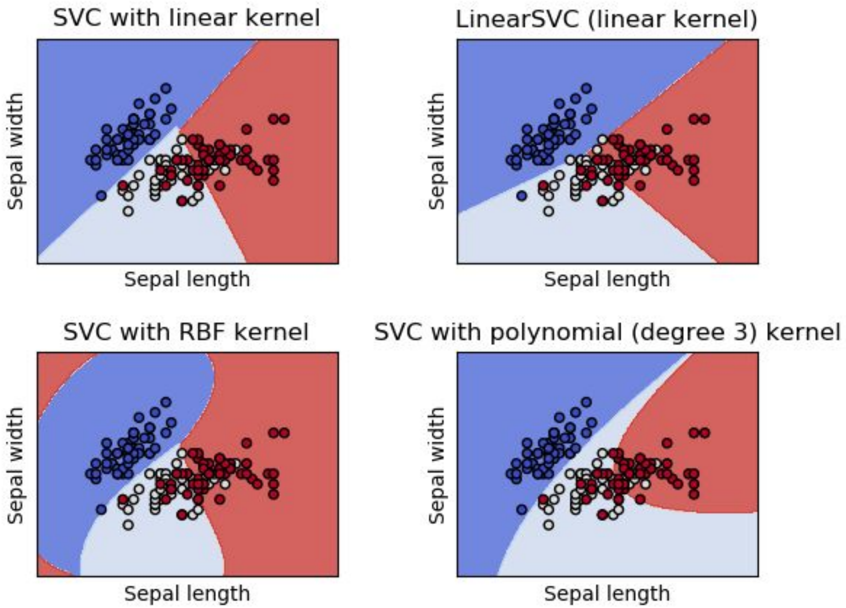
Features for SVM

SVM - light weight machine learning classifier, easily run on mobile devices

Statistical summary of signals:

FFT across x,y,z axes, max across all axes

Mean, sum, min, max, 1st derivative, media std, range, spectral band ratios

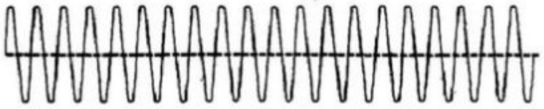


Modulation

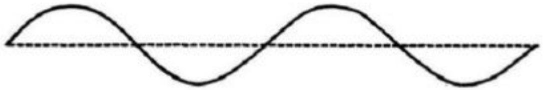
Inaudible. Human hearing 20Hz - 20kHz

Phones sample music at 44.1 - 48kHz

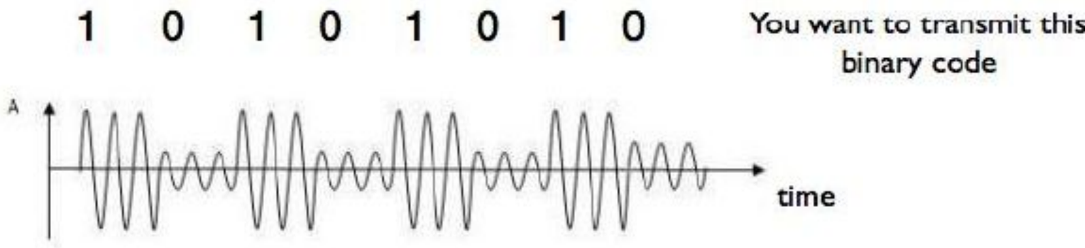
200Hz selected as carrier frequency - because inaudible



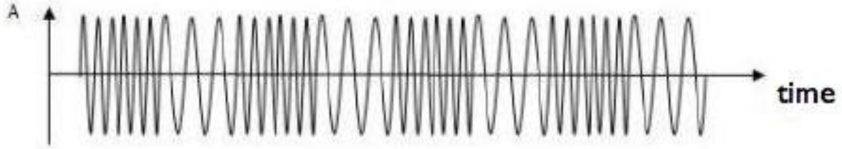
A CARRIER



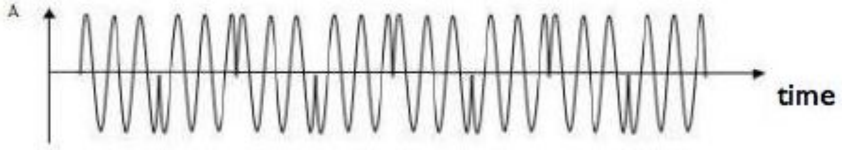
B SIGNAL



Amplitude modulation

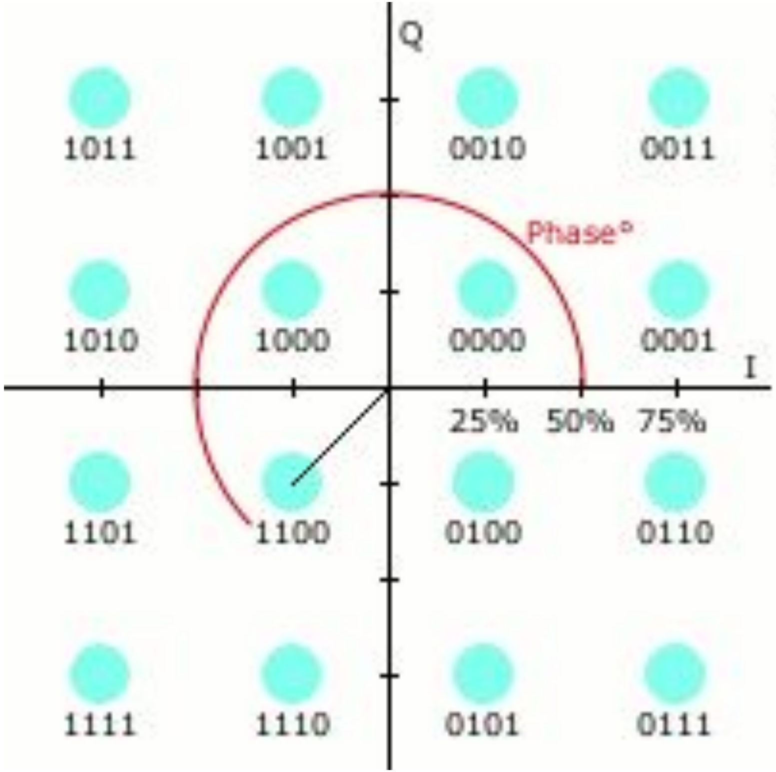


Frequency modulation



Phase modulation

QAM modulation



Amp	Phase	Data
25%	225°	1100

Bitstring corresponds to an amplitude and phase value

Combine amplitude and phase values and modulate onto carrier wave

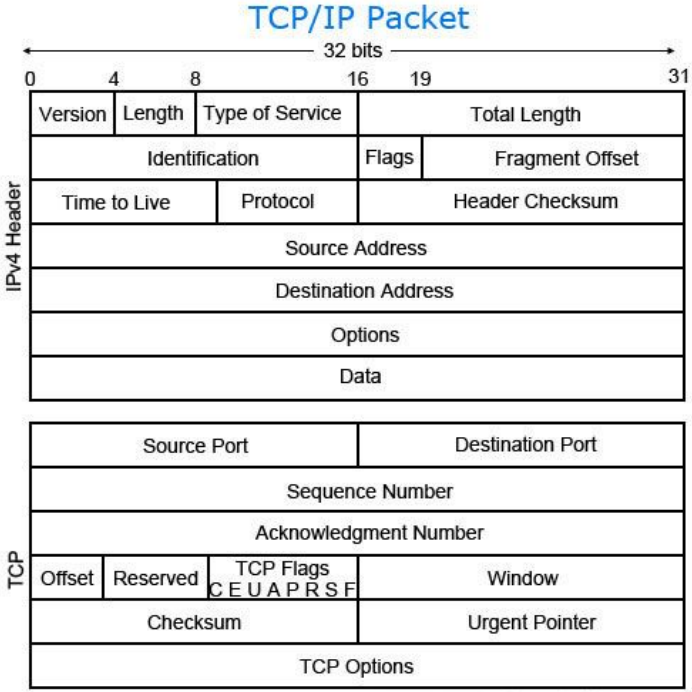
$$I(t)\cos(2\pi f_0 t)\cos(2\pi f_0 t) - Q(t)\sin(2\pi f_0 t)\cos(2\pi f_0 t)$$

Packetization

Data packetizations: preamble + data. Preamble allows receiver to know when a packet has arrived and to be synchronized with the incoming samples

Error detection: CRC - cyclic redundancy check, like a checksum
Put contents of packet through CRC algorithm

Error correction: Reed-solomon code, added redundancy to withstand bit errors


TCP/IP Packet

32 bits

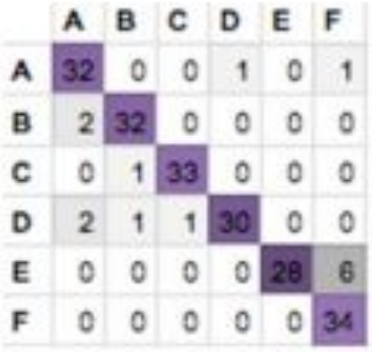
0 4 8 16 19 31

IPv4 Header

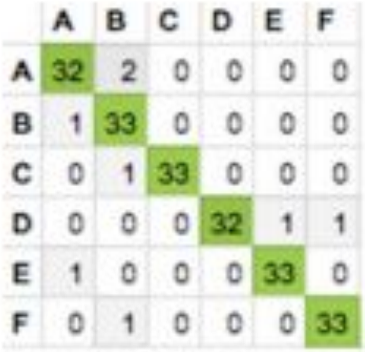
Version | Length | Type of Service | Total Length
Identification | Flags | Fragment Offset
Time to Live | Protocol | Header Checksum
Source Address
Destination Address
Options
Data

TCP

Source Port | Destination Port
Sequence Number
Acknowledgment Number
Offset | Reserved | TCP Flags C E U A P R S F | Window
Checksum | Urgent Pointer
TCP Options



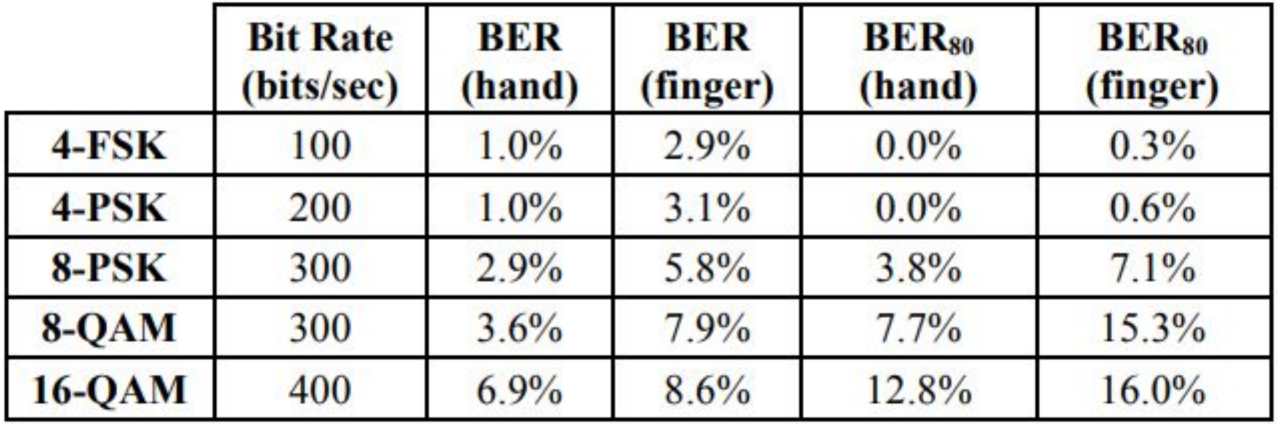
	A	B	C	D	E	F
A	32	0	0	1	0	1
B	2	32	0	0	0	0
C	0	1	33	0	0	0
D	2	1	1	30	0	0
E	0	0	0	0	28	6
F	0	0	0	0	0	34



	A	B	C	D	E	F
A	32	2	0	0	0	0
B	1	33	0	0	0	0
C	0	1	33	0	0	0
D	0	0	0	32	1	1
E	1	0	0	0	33	0
F	0	1	0	0	0	33

	A	B	C	D	E
A	33	1	0	0	0
B	3	30	0	1	0
C	1	0	32	0	1
D	0	0	0	34	0
E	1	1	1	0	31

Confusion matrix

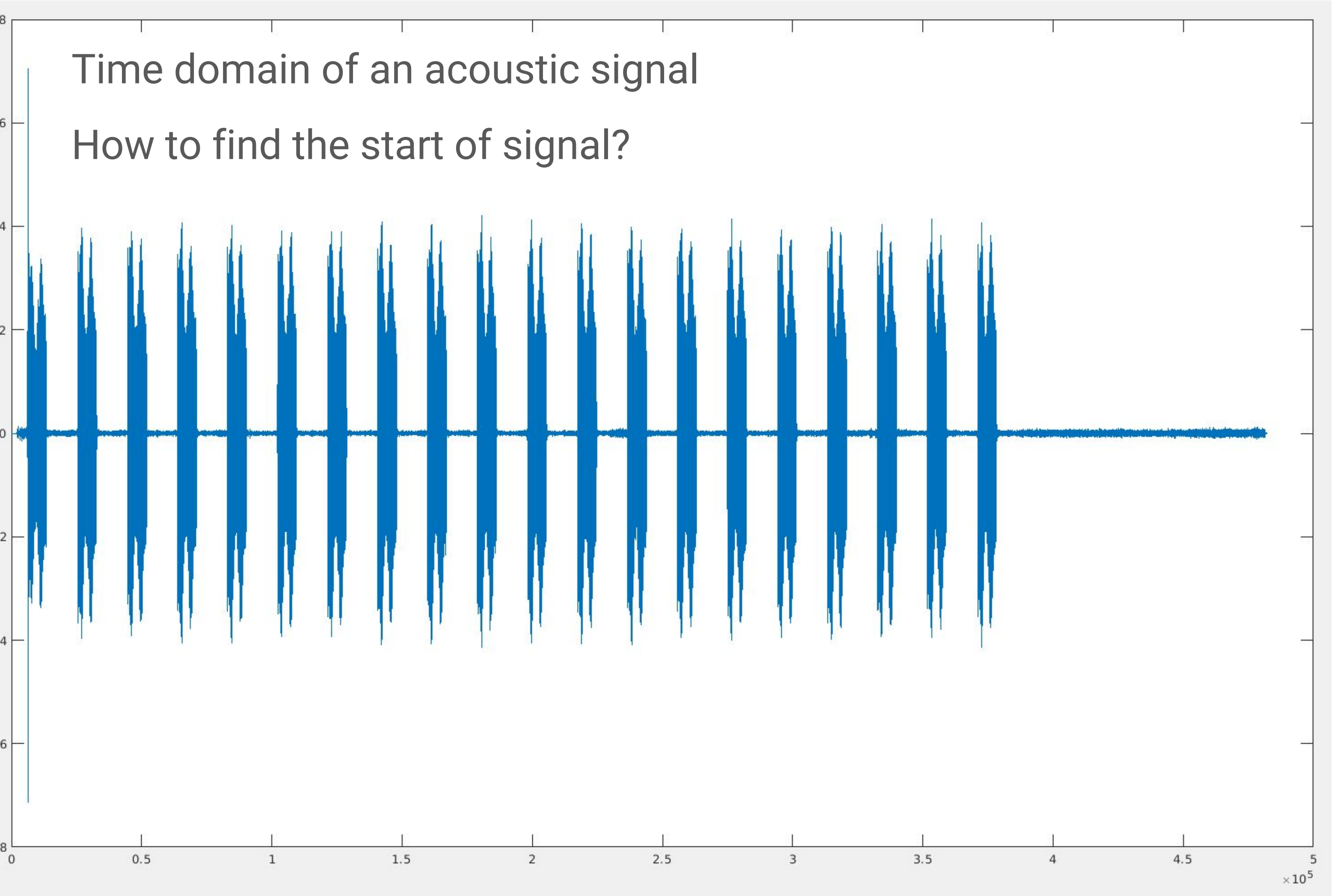


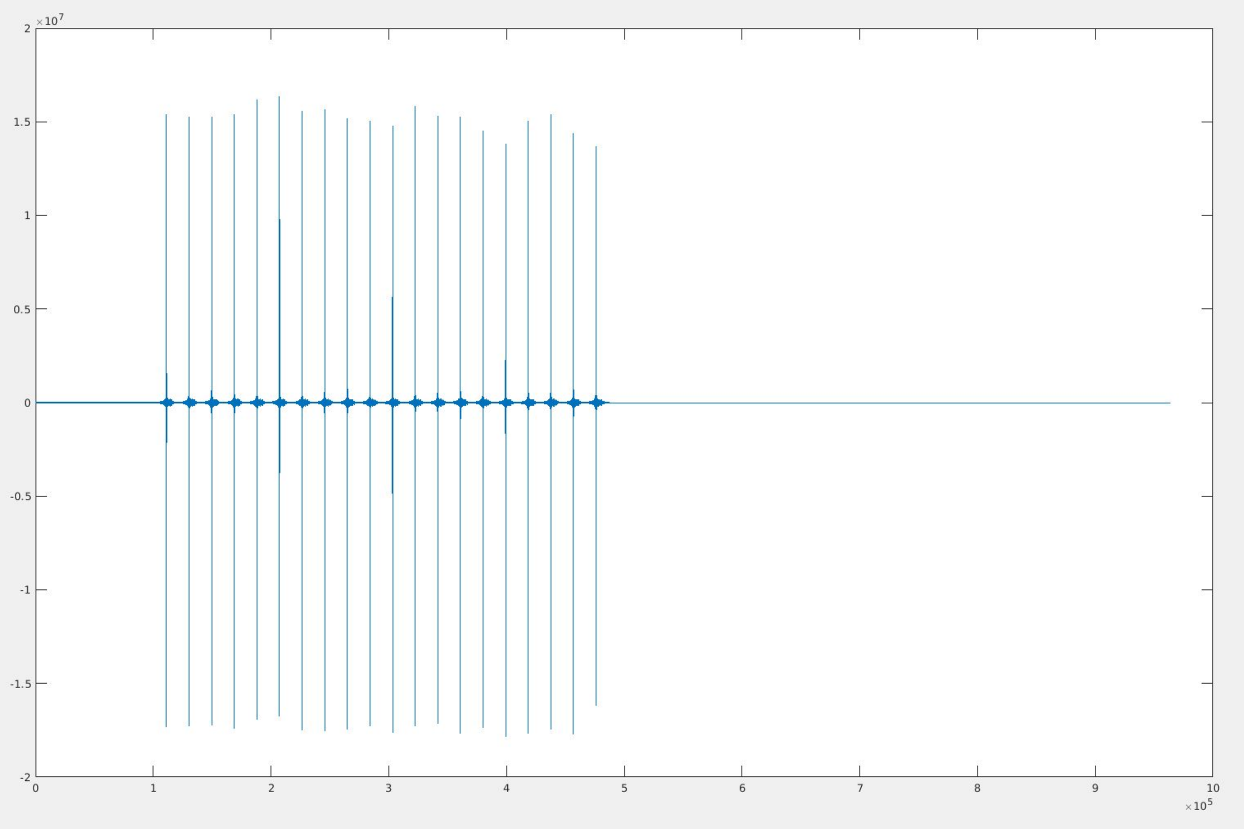
	Bit Rate (bits/sec)	BER (hand)	BER (finger)	BER_{80} (hand)	BER_{80} (finger)
4-FSK	100	1.0%	2.9%	0.0%	0.3%
4-PSK	200	1.0%	3.1%	0.0%	0.6%
8-PSK	300	2.9%	5.8%	3.8%	7.1%
8-QAM	300	3.6%	7.9%	7.7%	15.3%
16-QAM	400	6.9%	8.6%	12.8%	16.0%

Bit error rate
BER higher at higher bitrates

Time domain of an acoustic signal

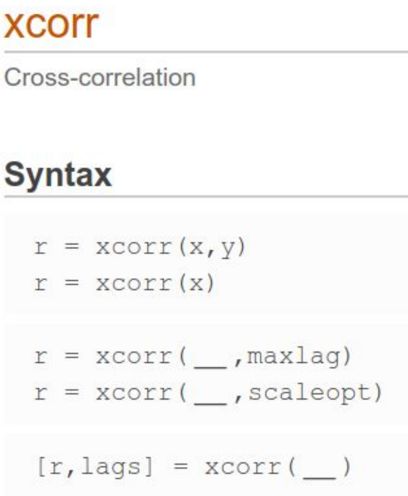
How to find the start of signal?





If we know the signal on the air we can correlate it with the known values.

Cross correlation will show peak indicating the starting sample.



xcorr

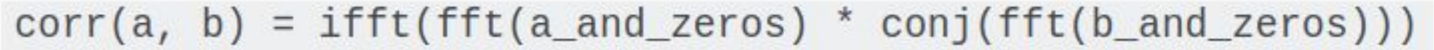
Cross-correlation

Syntax

```
r = xcorr(x,y)
r = xcorr(x)

r = xcorr(__,maxlag)
r = xcorr(__,scaleopt)

[r,lags] = xcorr(__)
```

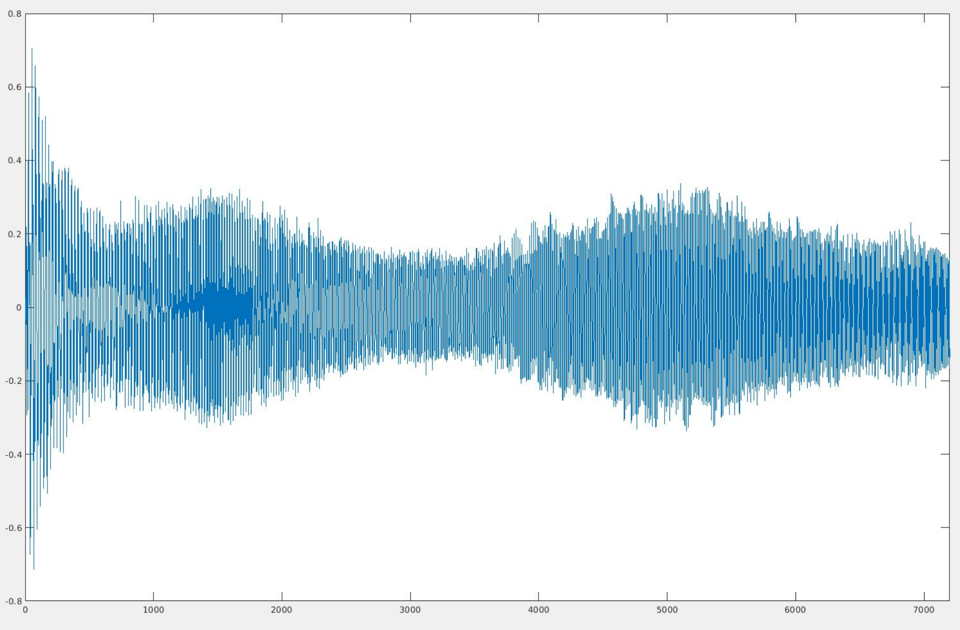


```
corr(a, b) = ifft(fft(a_and_zeros) * conj(fft(b_and_zeros)))
```

IFFT goes from frequency domain to time domain

Conjugate of complex number is real component and negated imaginary component

Individual signal



Time domain has 7200 samples

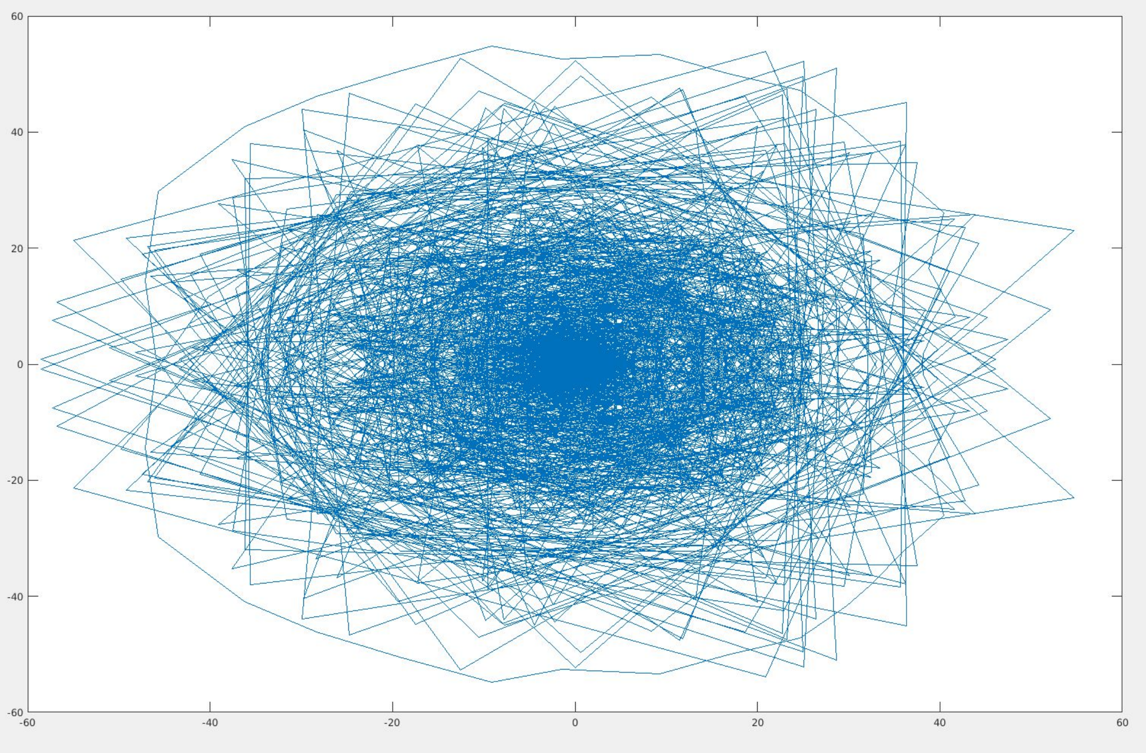
The signal is a chirp.

It increases linearly from 1.8 kHz to 4.4 kHz

Based on Nyquist theorem we need at least 8.8kHz sampling rate

It was sampled at 48kHz on a phone

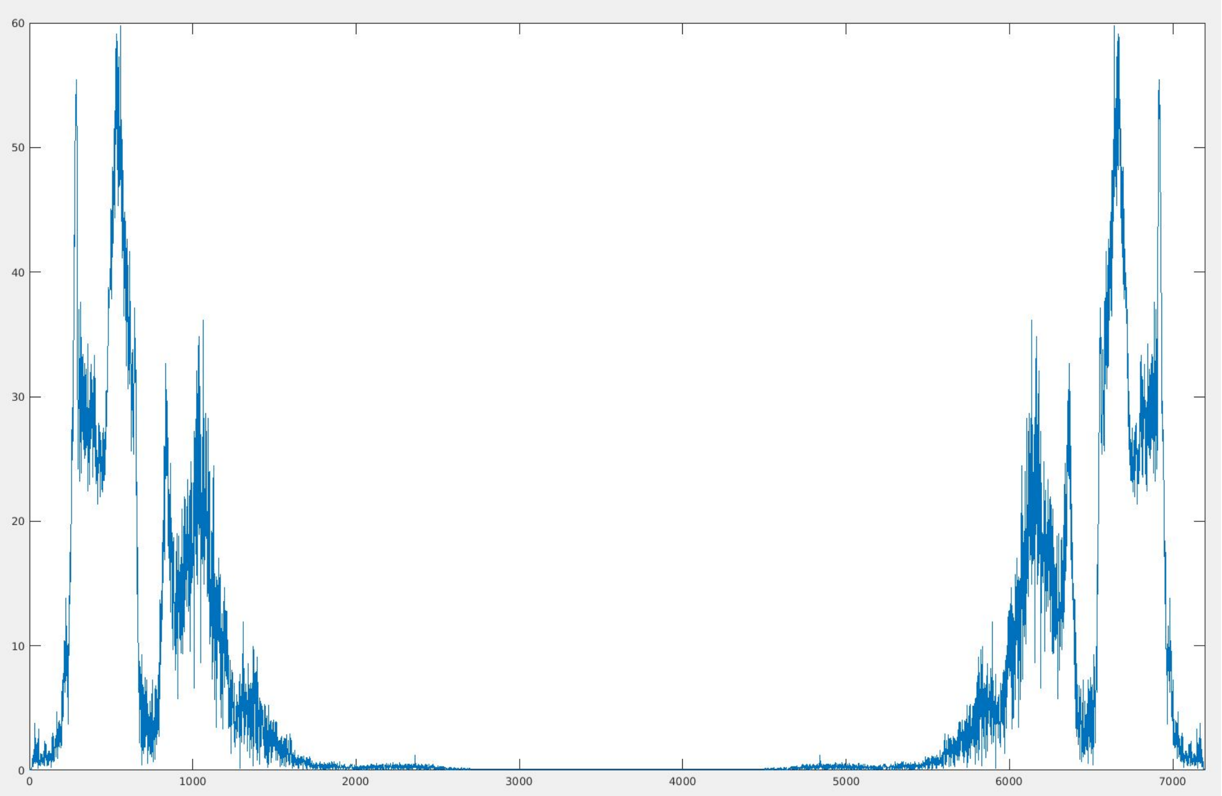
Complex FFT



The chirp had 7200 samples. So we can take at max a 7200 point FFT.

FFT returns a real and imaginary component

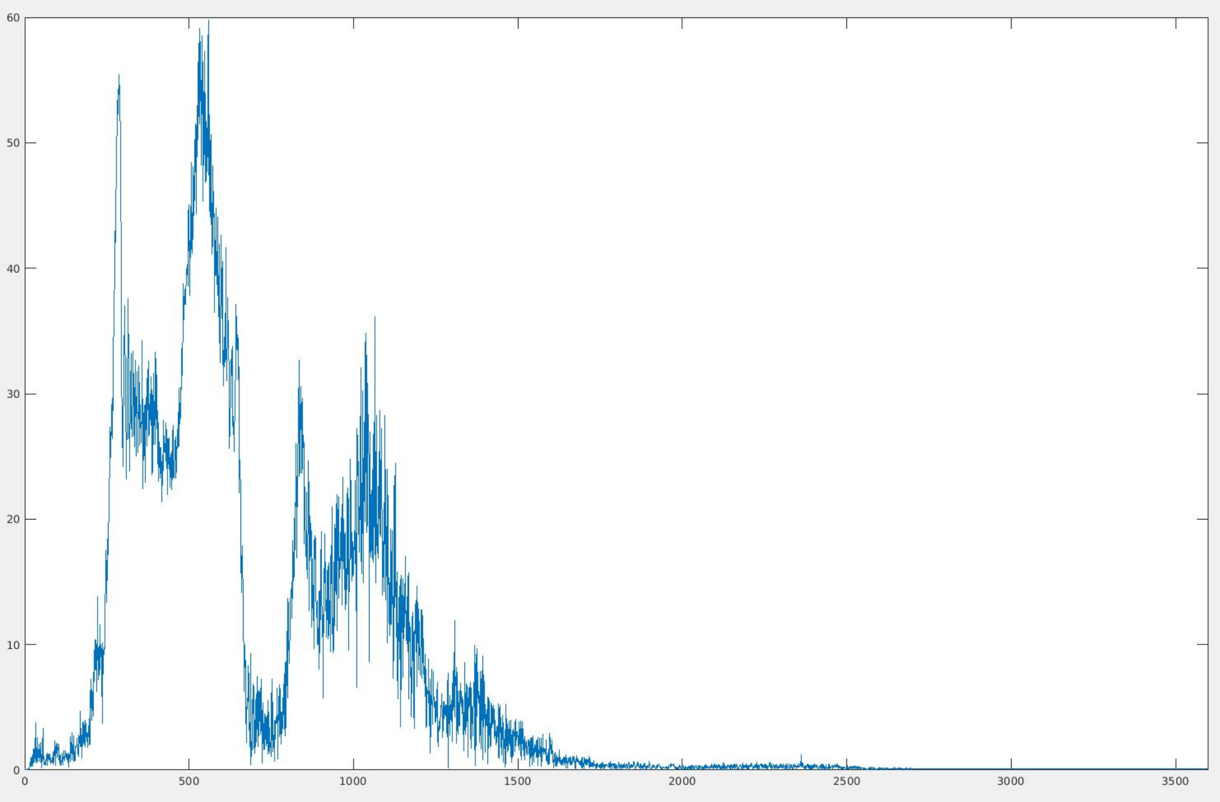
Two sided FFT



After taking absolute value we will get a two-sided FFT.

Discard the right hand side at the middle.

Single sided FFT



On the x axis is frequency.

On the y axis is amplitude

Note here that we have 3600 frequency bins on the x axis.

This corresponds to a range from 0 - 24kHz

Each bin represents the power for 24000/3600 = 6.67 Hz

FFT bin resolution

What if we want a smaller FFT bin/higher resolution?

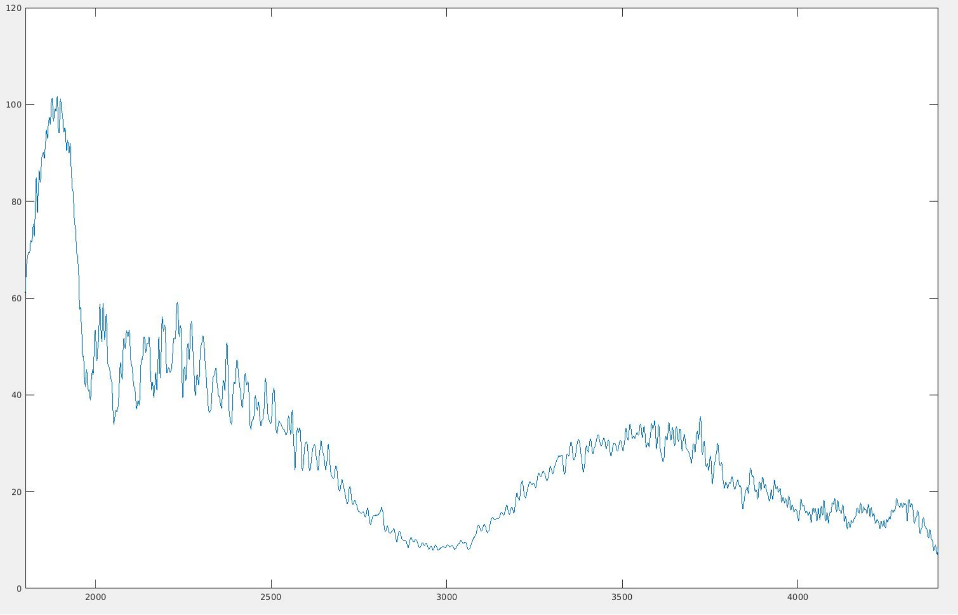
What if we did a 48000 point FFT? Then each bin would have a resolution of 1Hz

We can’t do this because our original signal was only 7200 long. If we did this, the FFT would simply interpolate values across bins, and we won’t actually get 1Hz of resolution.

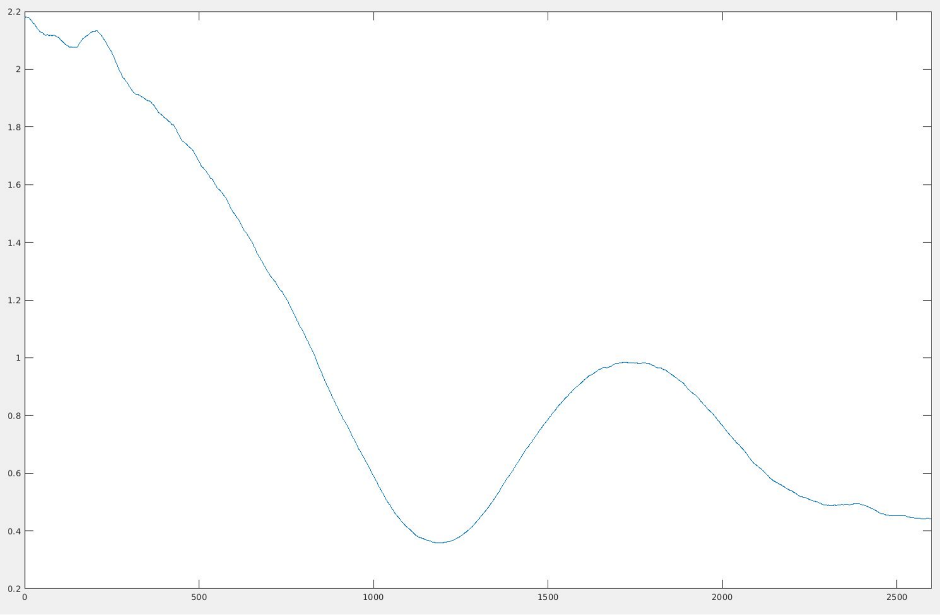
To resolve this take FFT over multiple chirps. Take over 7 chirps (7200*7=50400 samples), then perform 48000 point FFT, then we get 1Hz resolution.

High resolution is important for applications in tracking and localization

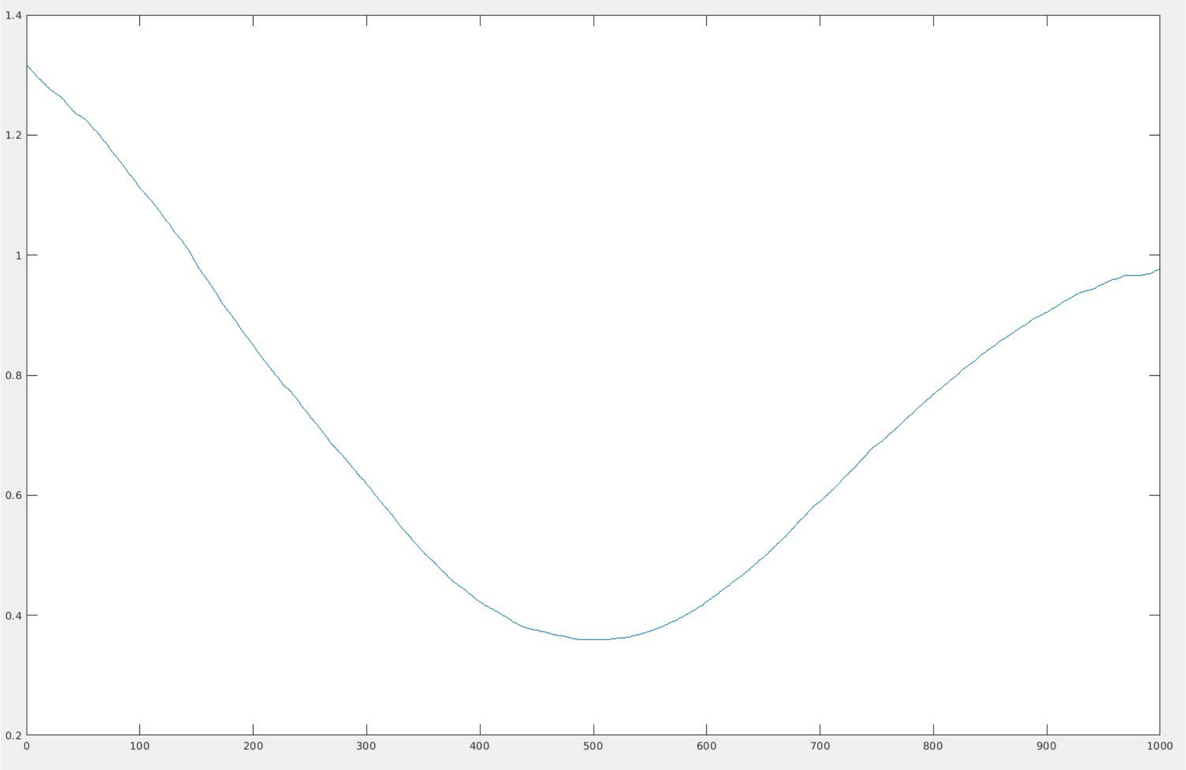
Isolate signal within resonant frequencies



Moving average smoothing



Find prominent peaks to get dip



Pass into machine learning classifier as a vector of 1000 features

Machine learning training

State of the art machine learning techniques need loads of data. The more open-ended or complex the problem, the more data is needed

Ground truth is difficult to label at scale

Most interactive projects - not enough data

Need: powerful ML techniques that require only few examples and infer more from the context of use

OMNITOUCH

Image processing primer

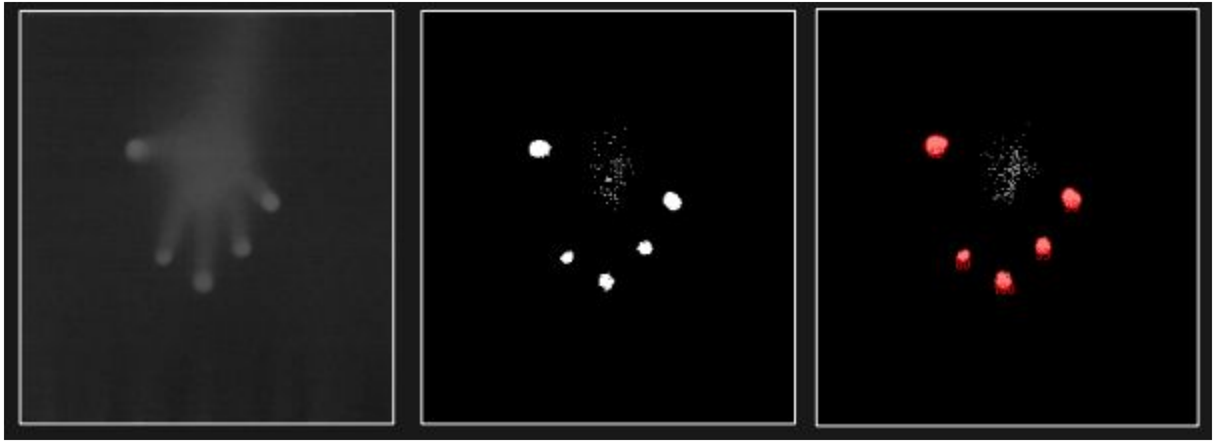
Typical camera pipeline

1. Segment foreground from background (edge detection)
2. Cluster foreground pixels into blobs
3. Track blobs over time

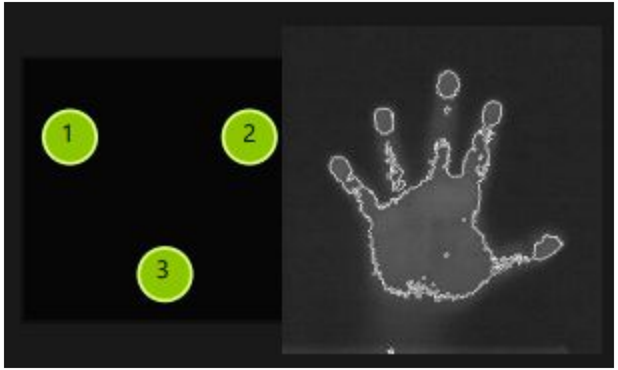
Image segmentation

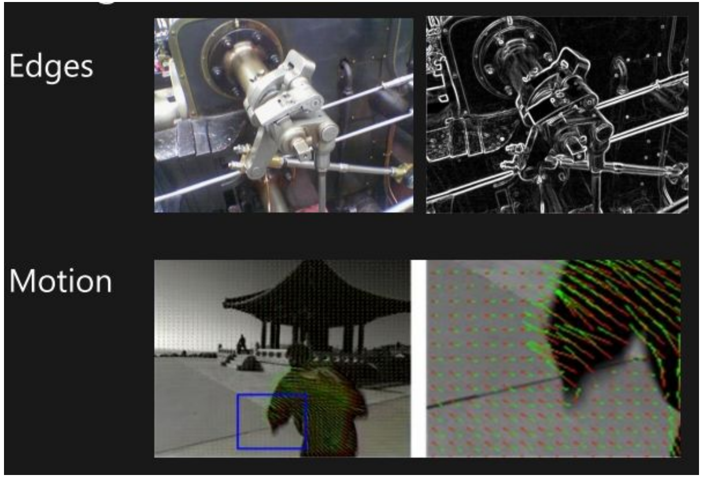
Thresholding based on intensity (or color)

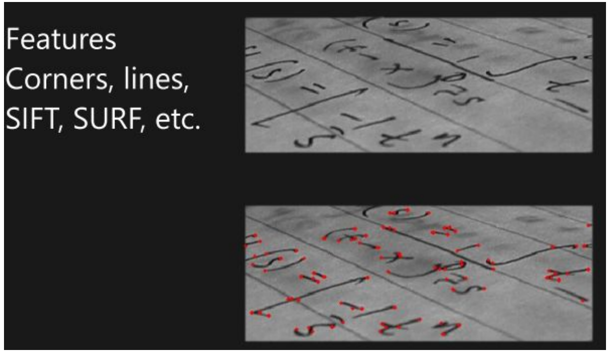
Blob tracking



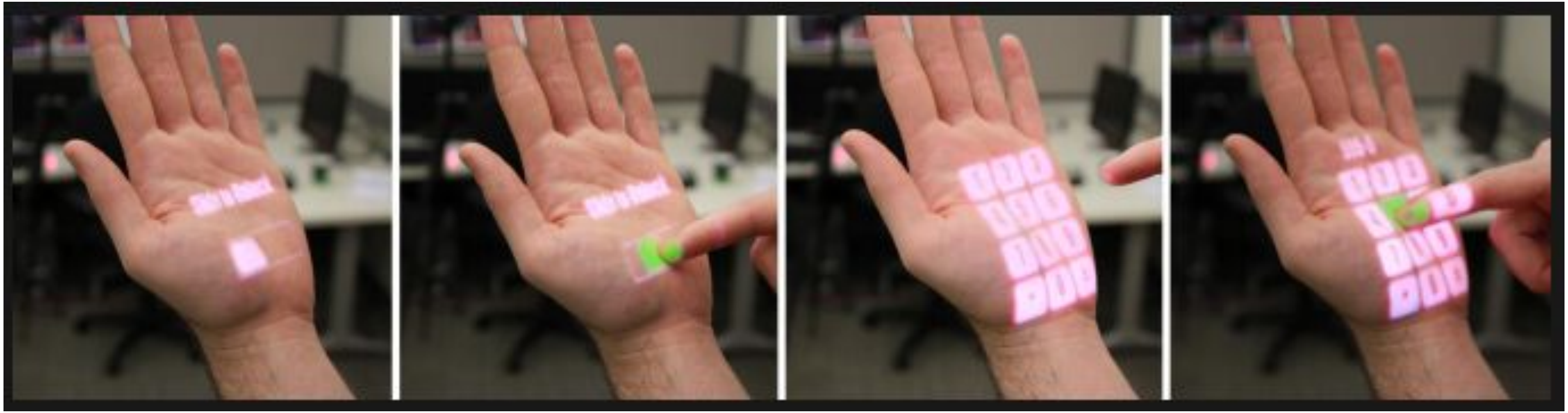
Pulling out discrete contact points is an ill-posed problem





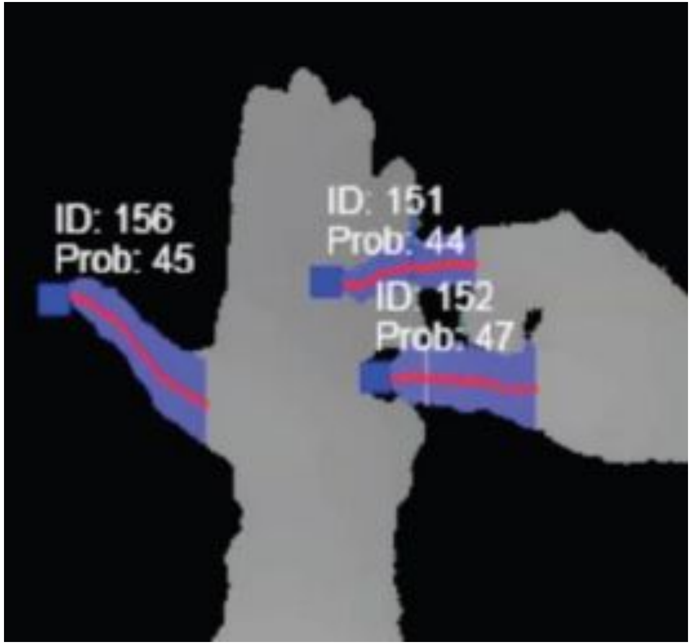


Turn every surface into a touchpad



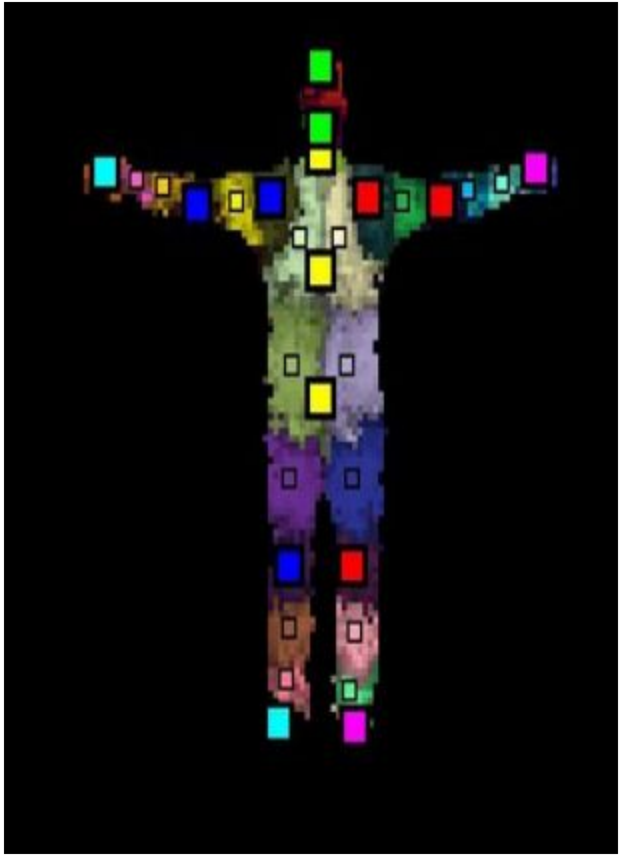
Tracking high-level constructs (fingers)

- Take only the ends of objects with physical extent (“fingertips”)
- Detect contact (“click”)
- Refinement of position while clicked (“drag”)



Skeleton tracking (Kinect)

1) Classify each pixel's probability of being each of 32 body parts
2) Determine probabilistic cluster of body configurations consistent with those parts
3) Present the most probable pose to the user

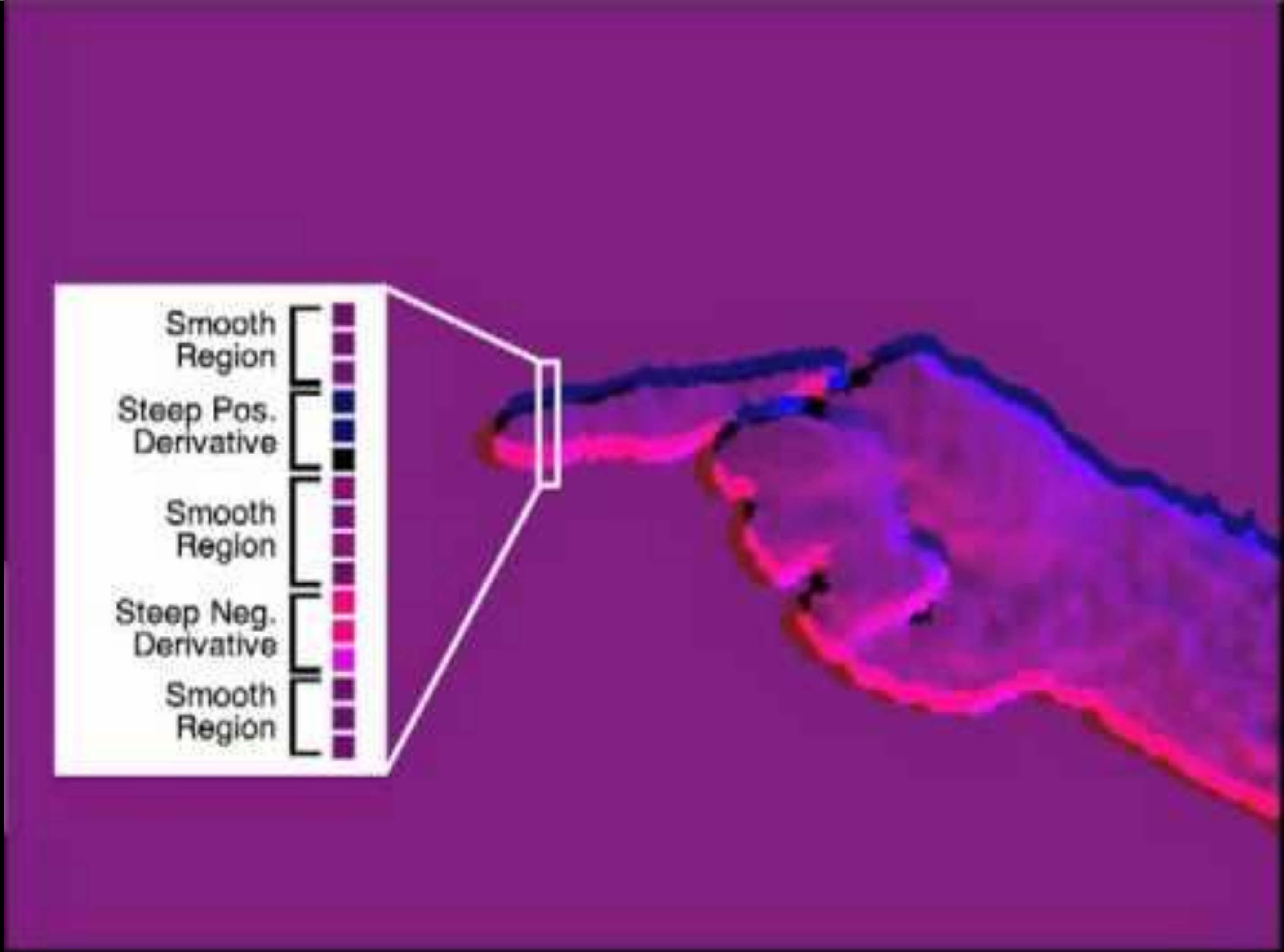


Hand tracking

Leap motion controller:

2 IR cameras

3 IR LEDs


Smooth Region
Steep Pos. Derivative
Smooth Region
Steep Neg. Derivative
Smooth Region

Omnitouch

Shoulder-worn system, no instrumentation of user or environment

Use anything as a surface

320x240 depth map at 30fps, images as close as 20cm, depth error of 5mm

Resolve x,y,z position and whether they are touching or hovering over surface

1) Depth map, compute depth derivative using sliding window
2) Look for vertical slices of cylinder-like objects using dynamic parameters
3) Isolate candidate fingers
4) Greedily proximate solices into contiguous paths
5) Fingertip positions smoothed with Kalman filter

Click algorithm

Compute midpoint of finger path

Flood fills to detect finger clicks, determines irregular states and clicks and drags

If pixel threshold is passed, then the finger is determined to be clicked

Hysteresis applied to reduce flickering

Surface algorithm

Distinct surfaces segmented for projection in front of user

Distinct surfaces segmented by performing 3D connected components operation. Surfaces smaller than hand are discarded

Compute orientation wrt camera

POSIT algorithm to find position and orientation of projector - one time calibration based on field of view, center of projection and calibration points

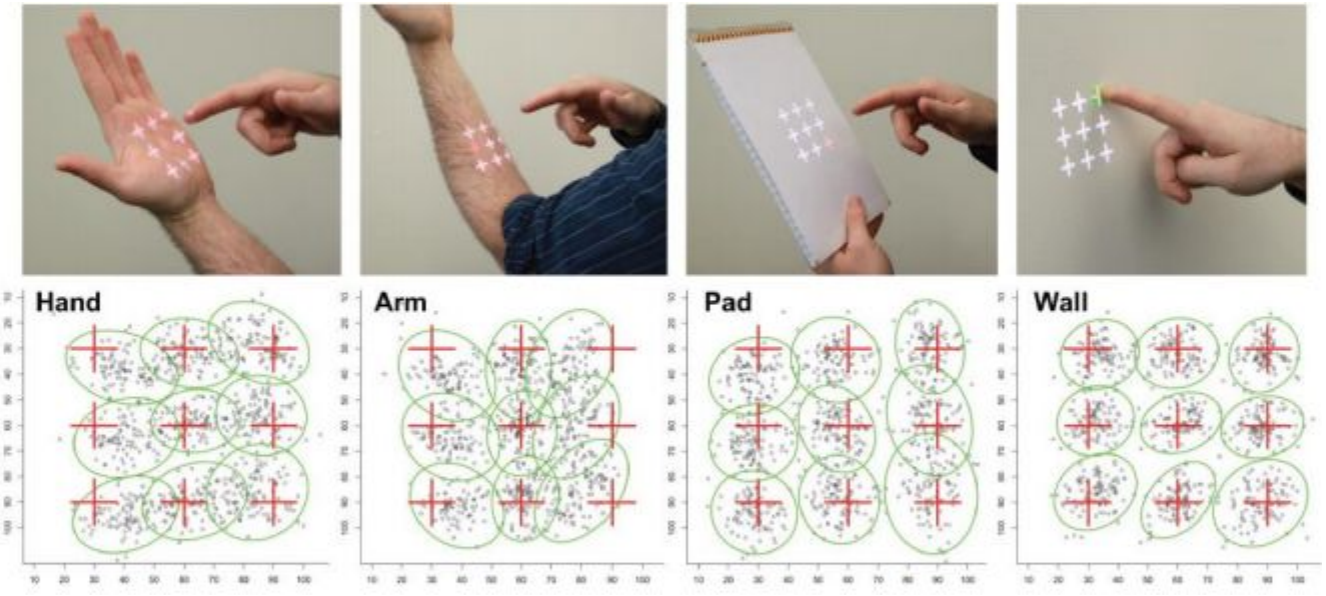
Defining interactive spaces

Hand-sized interfaces. Downside: limited size, sub-optimal centering.

Performs surface classification and automatically sizes, positions and tracks an interface given the projection area(hand, arm, pad, wall, table). Downside: unscalable

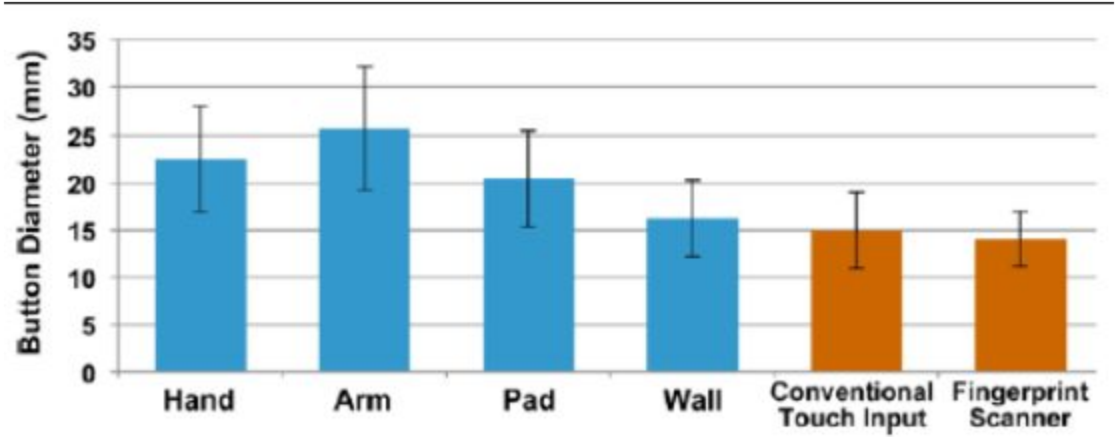
User-specified placement: click on surface and center location

6048 click trials

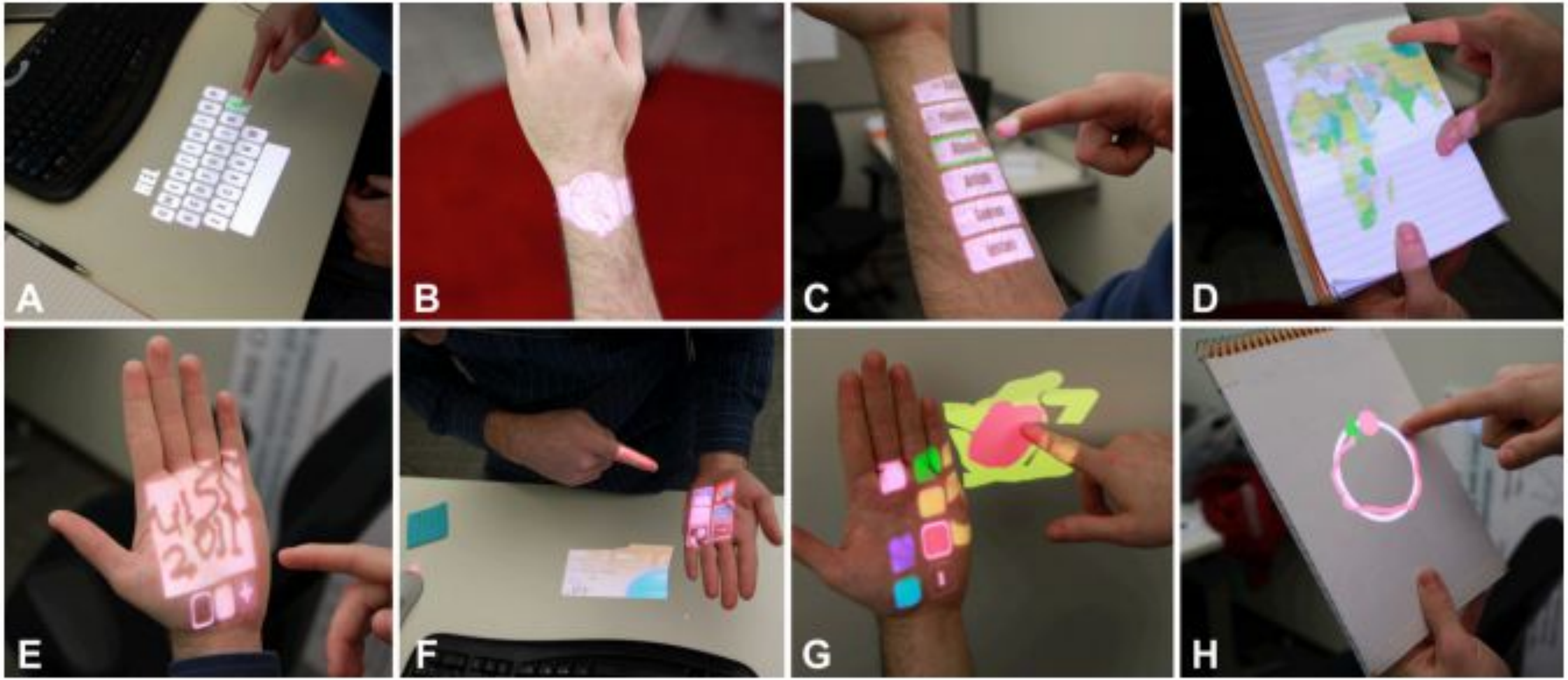


Click spatial accuracy

With 0.5s timeout rejection ~ 98.9% click accuracy



Applications



Questions to discuss

- Technical pros and cons of each of the approaches
- Examine both papers through Belotti’s question framework