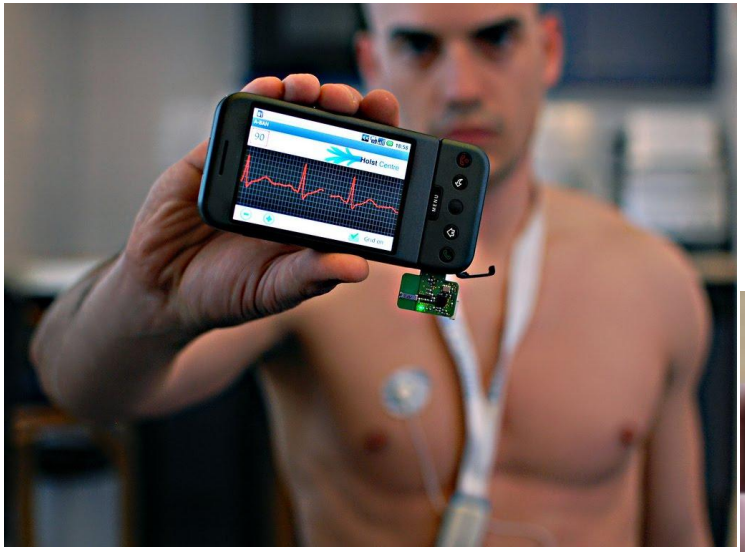
AccelPrint: Imperfections of Accelerometers Make Smartphones Trackable

Sanorita Dey, Nirupam Roy,
Wenyuan Xu, Romit Roy Choudhury, Srihari Nelakuditi

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

UNIVERSITY OF
SOUTH CAROLINA

People use hundreds of apps

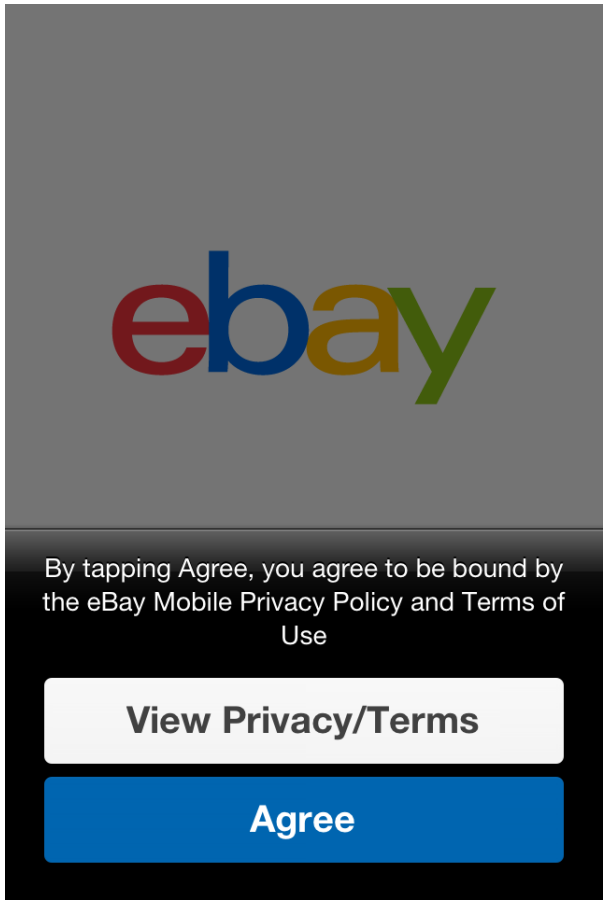


Some apps are sneaky

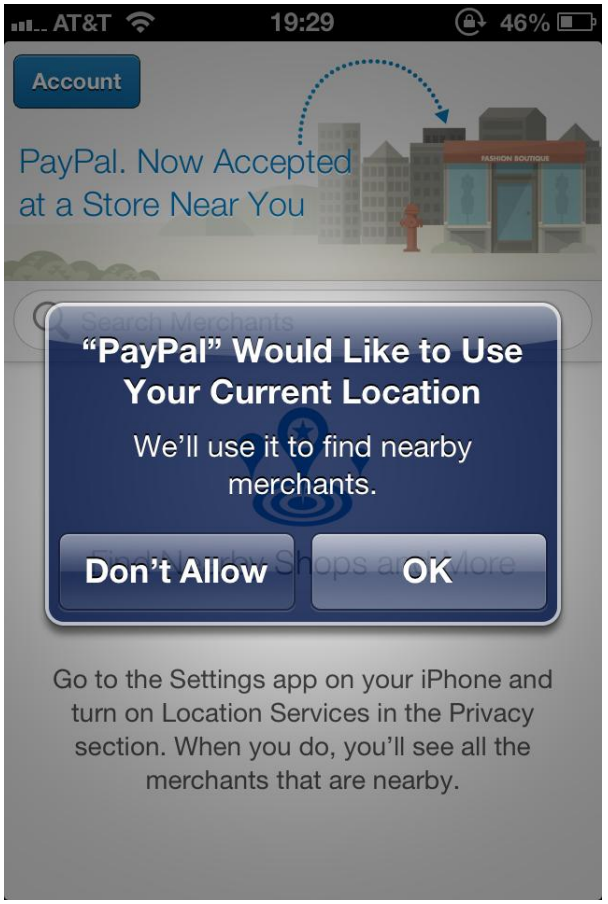
- Exchanging IDs without consent is rampant
  - IMEI (device id), IMSI (subscriber id), or ICC-ID (SIM card serial number) help track users
- One possible Solution: TaintDroid
  - Realtime filtering of exchange of device IDs

Law: Get user's consent

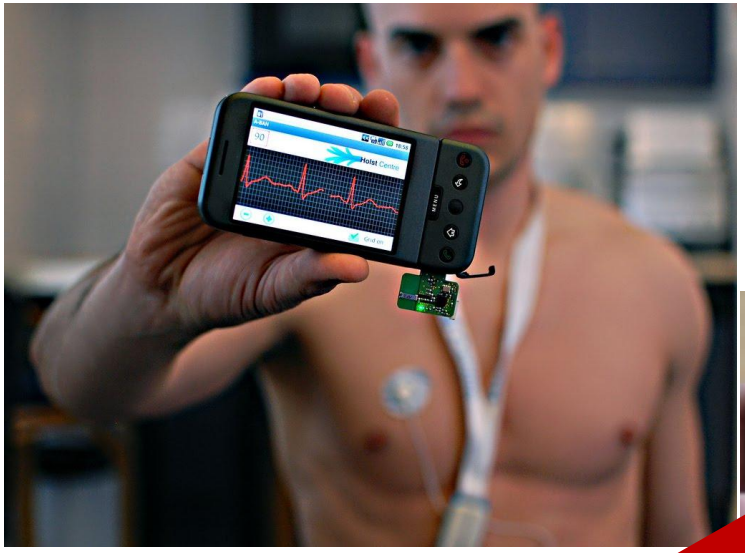
- While installing a cookie



- While sharing location



People use hundreds of apps



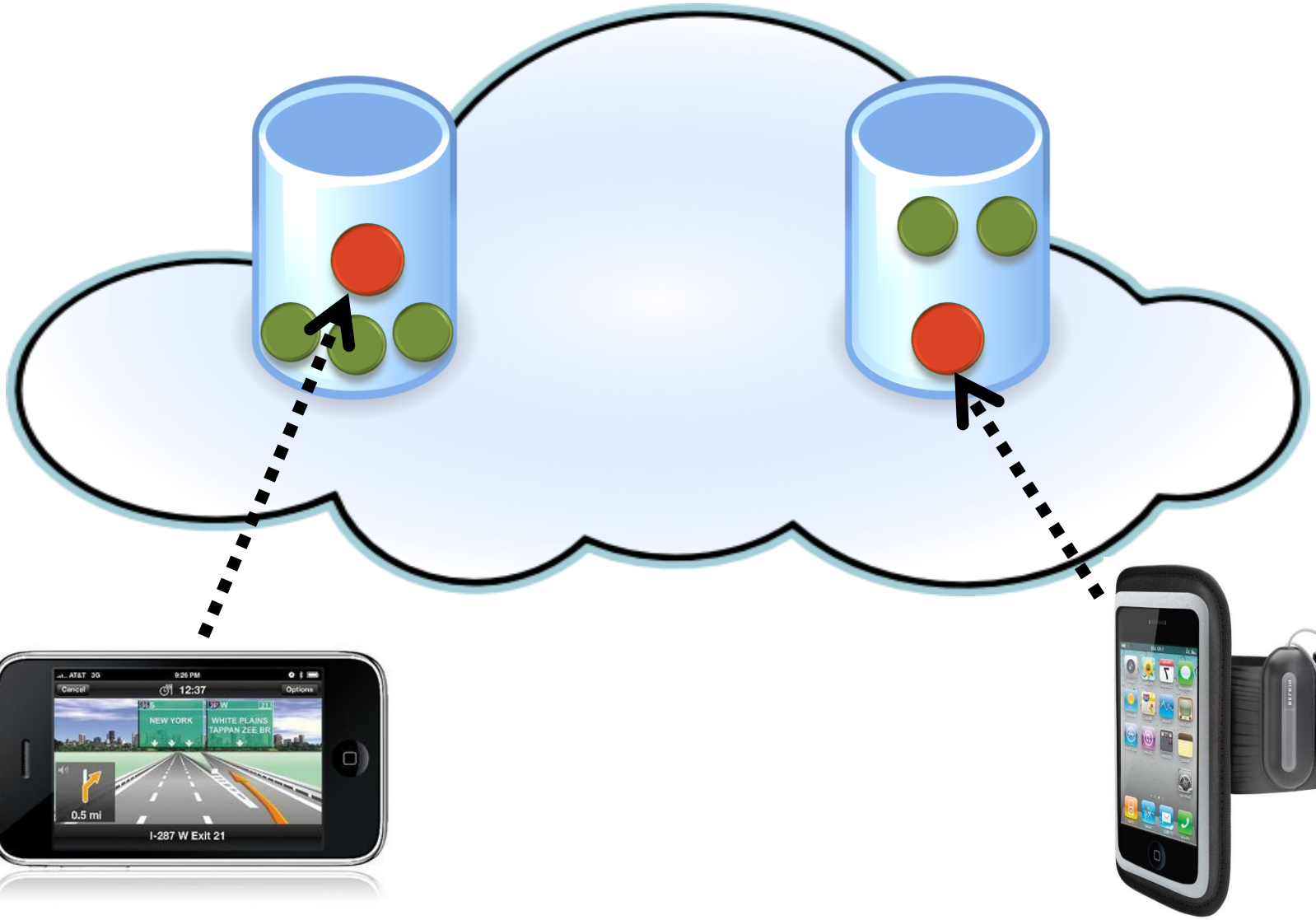
Is it safe to expose accelerometer data?

Our findings

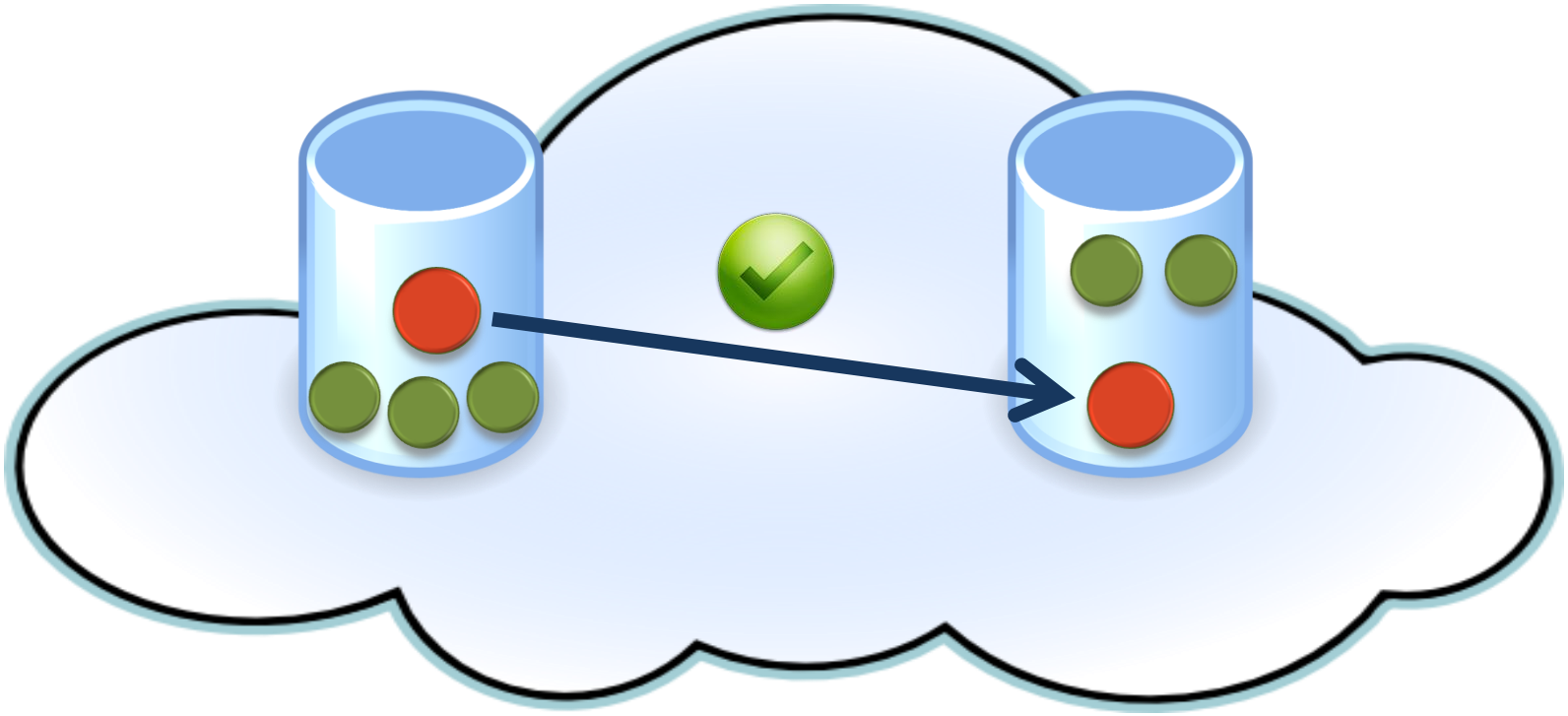
Other sensors can also potentially track the users

Accelerometers have fingerprint

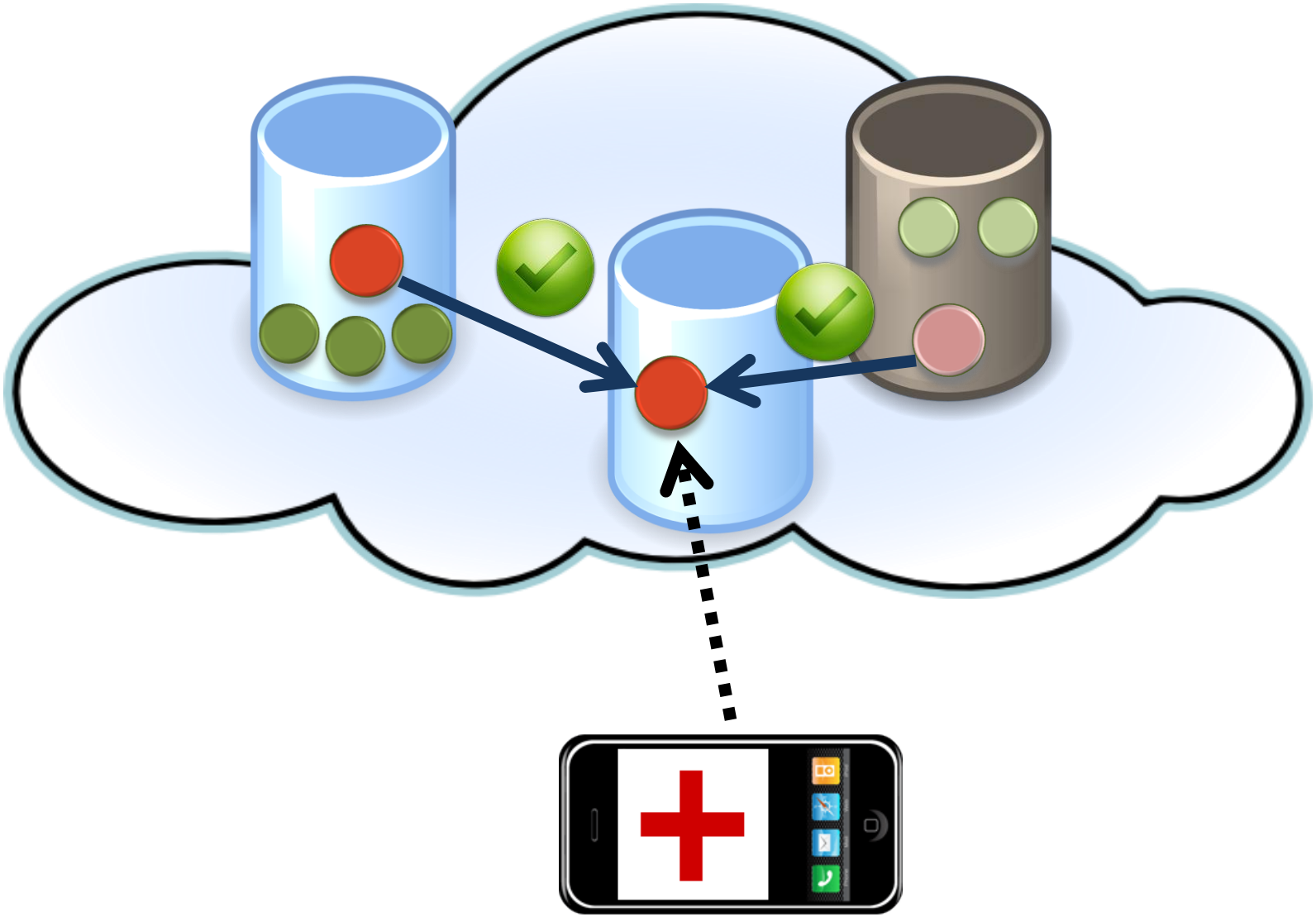
What if accelerometers have fingerprints?



What if accelerometers have fingerprints?



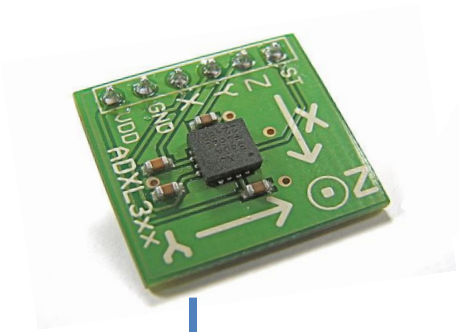
What if accelerometers have fingerprints?



Evidence of fingerprint

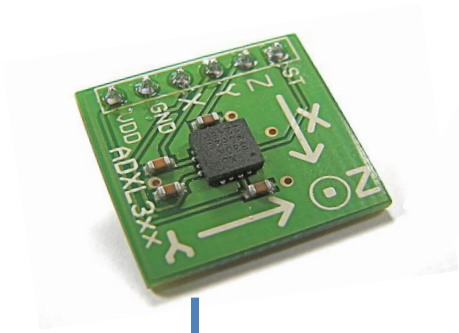
Toy Experimental Setup

Controlled, Identical Impetus



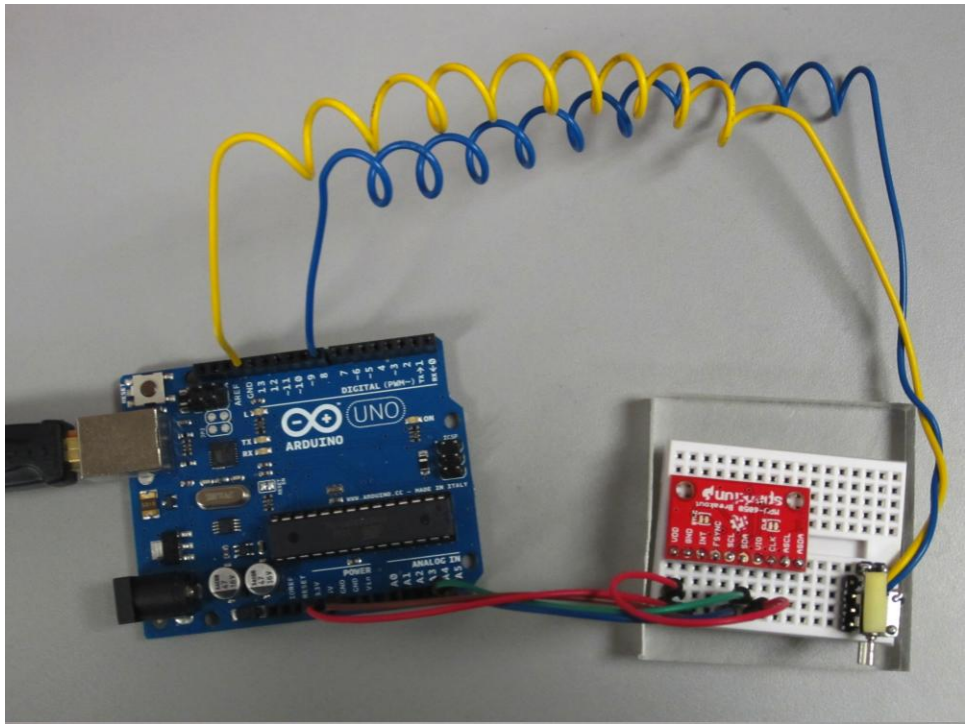
...

Toy Experimental Setup

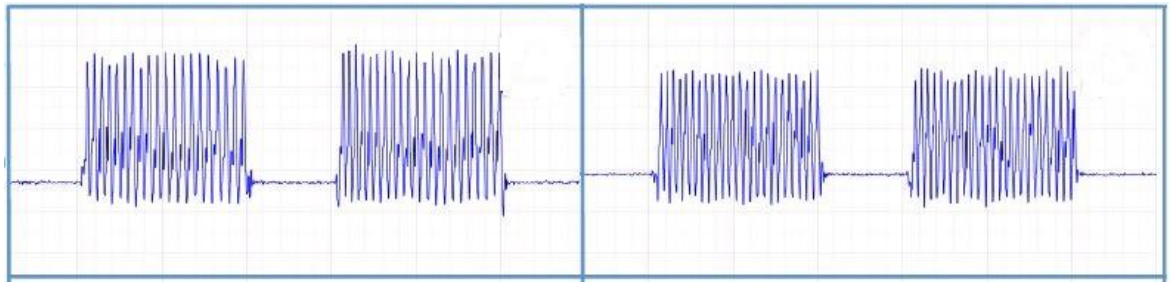


Toy Experimental Setup

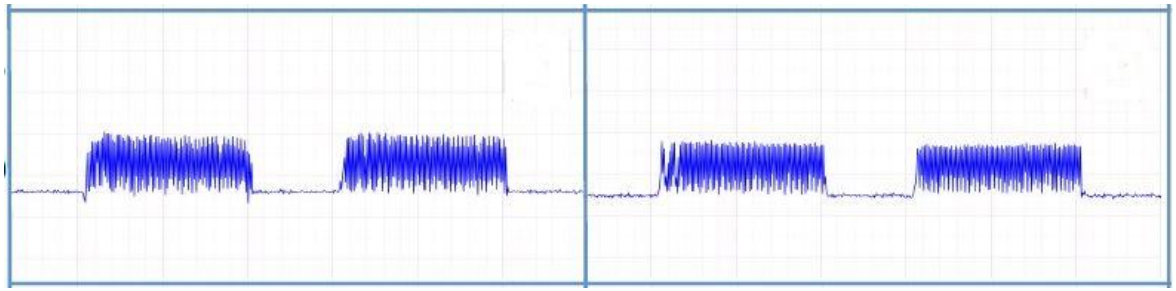
- Six stand-alone accelerometer chips
- Stimulation with an external vibration motor
- Arduino to control vibration and collect accelerometer readings



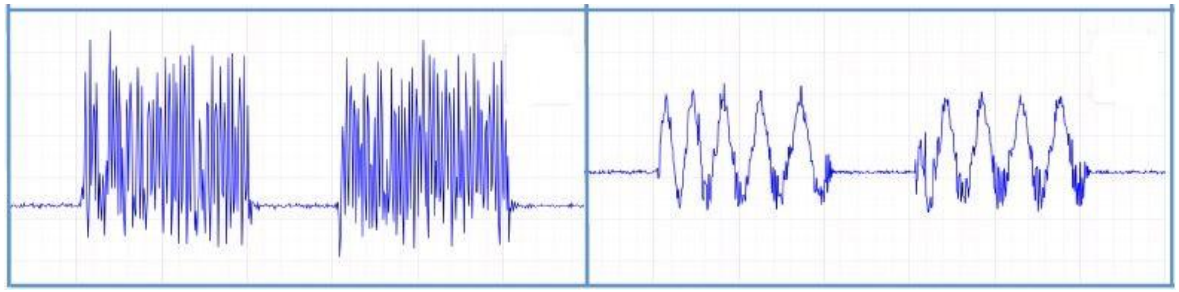
Accelerometers are distinguishable



Accelerometer chips of Samsung Galaxy S3

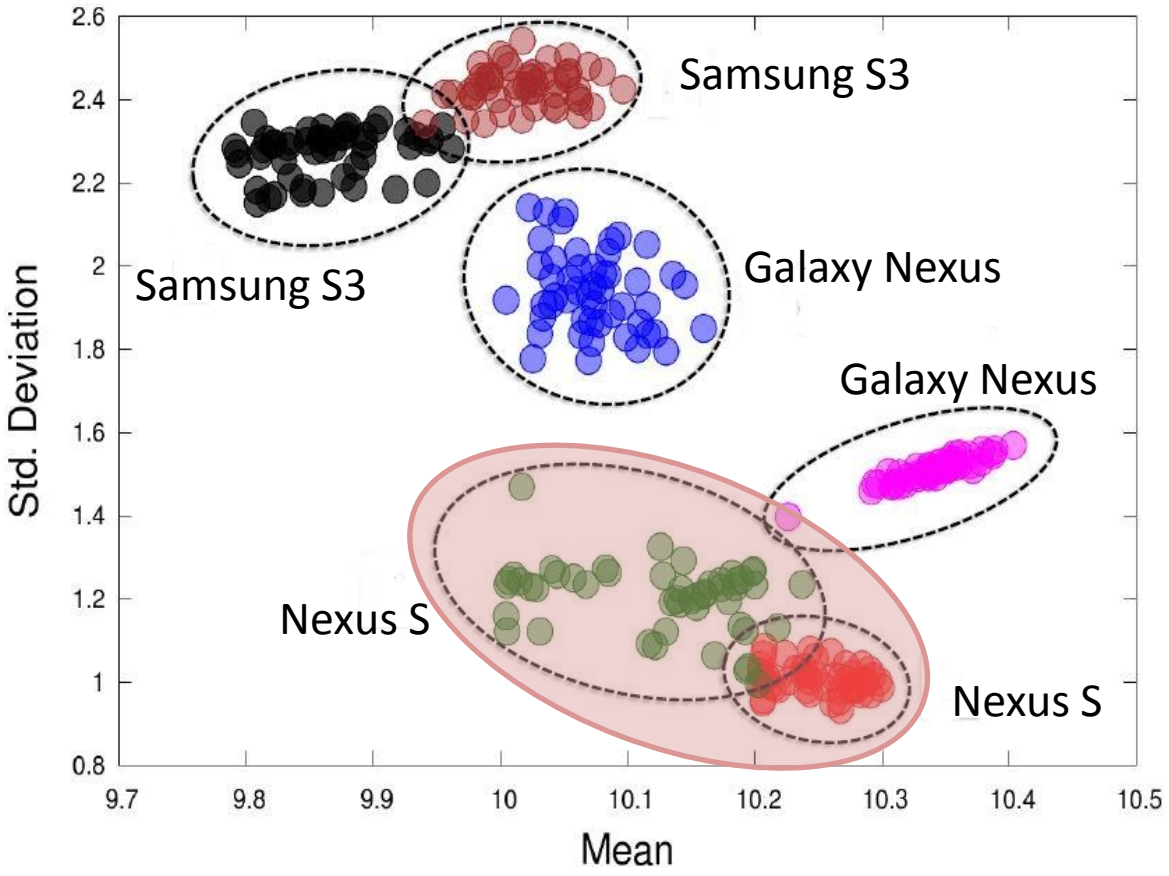


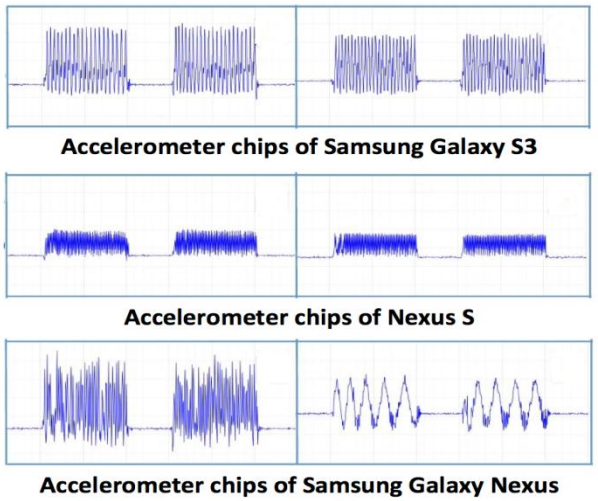
Accelerometer chips of Nexus S



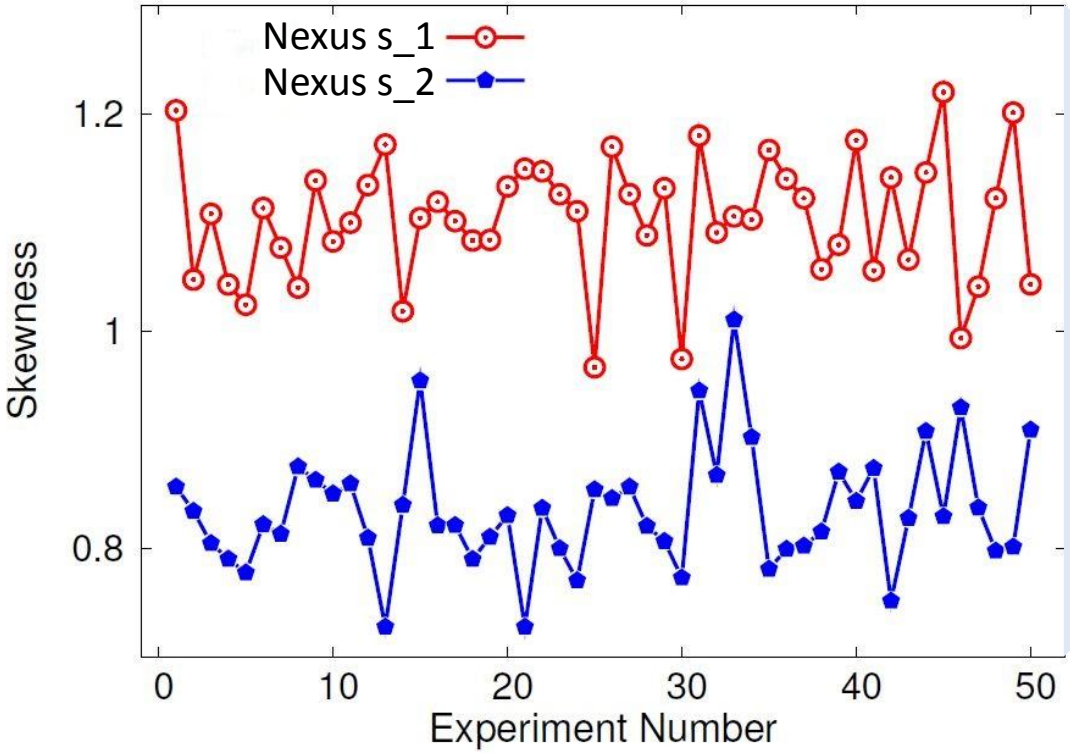
Accelerometer chips of Samsung Galaxy Nexus

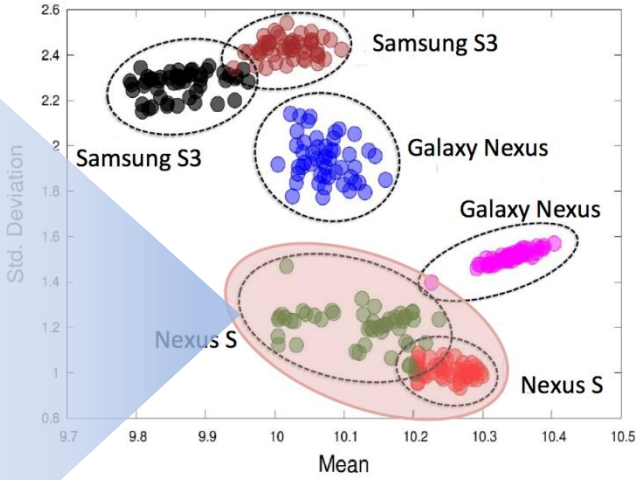
Accelerometers are distinguishable

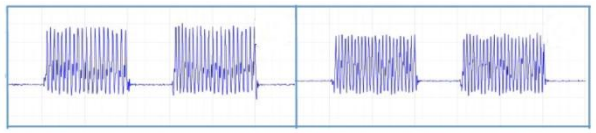




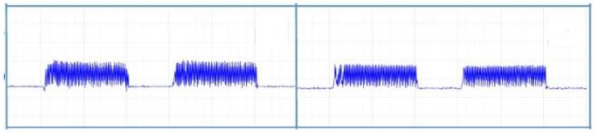
Accelerometers are distinguishable



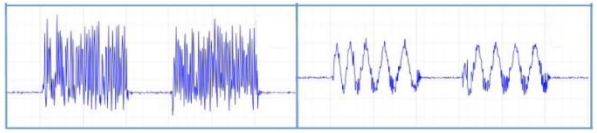




Accelerometer chips of Samsung Galaxy S3



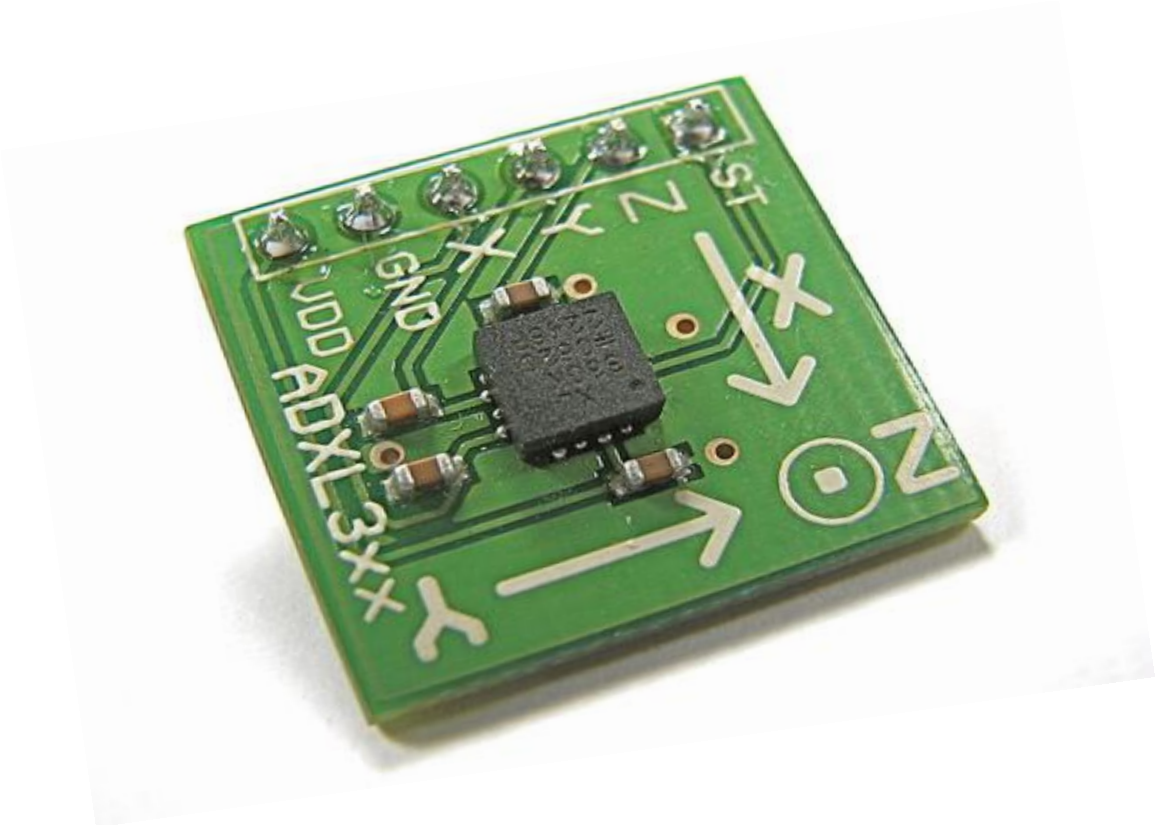
Accelerometer chips of Nexus S



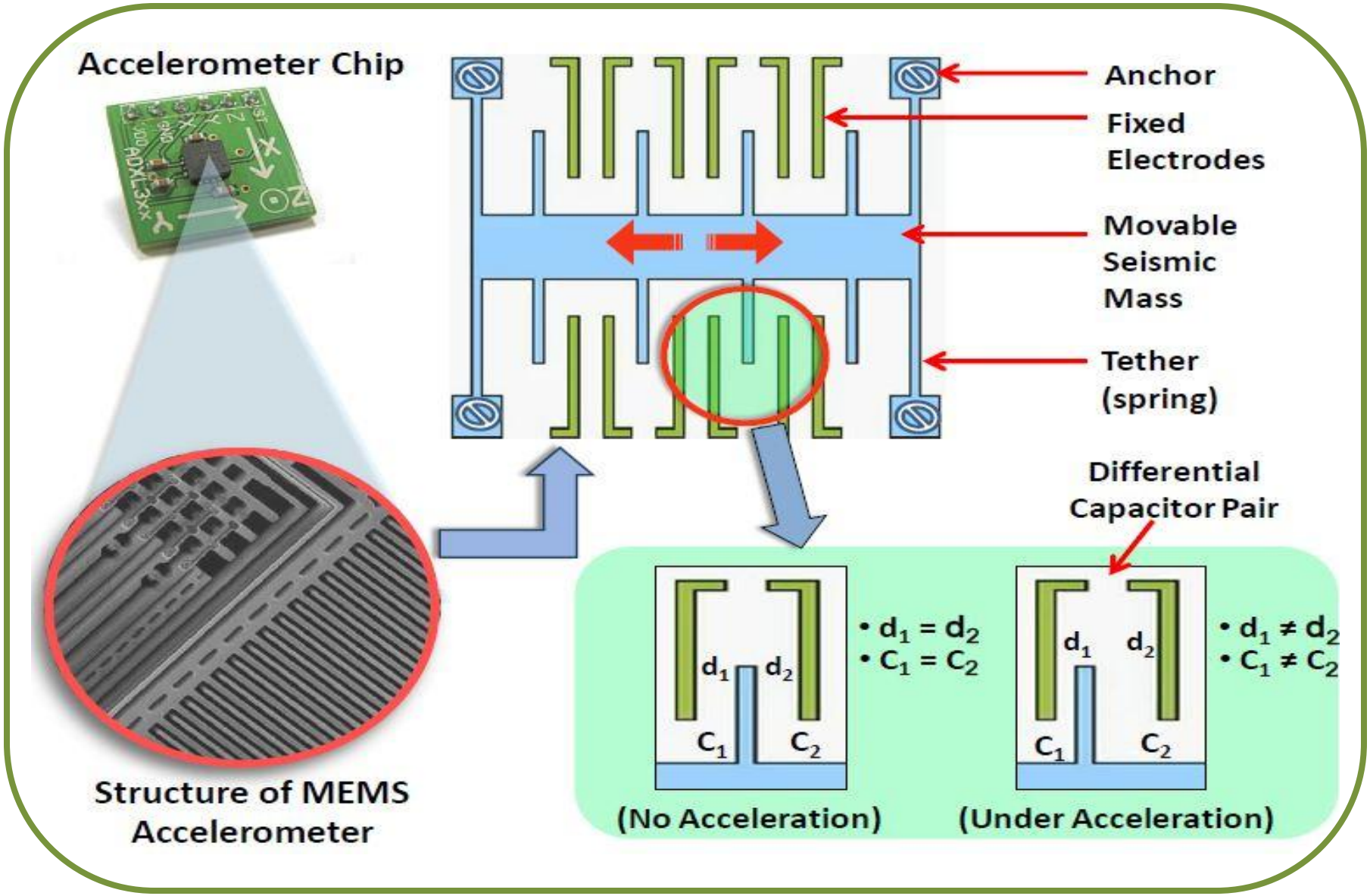
Accelerometer chips of Samsung Galaxy Nexus

Why are accelerometers distinct?

Accelerometers are based on MEMS

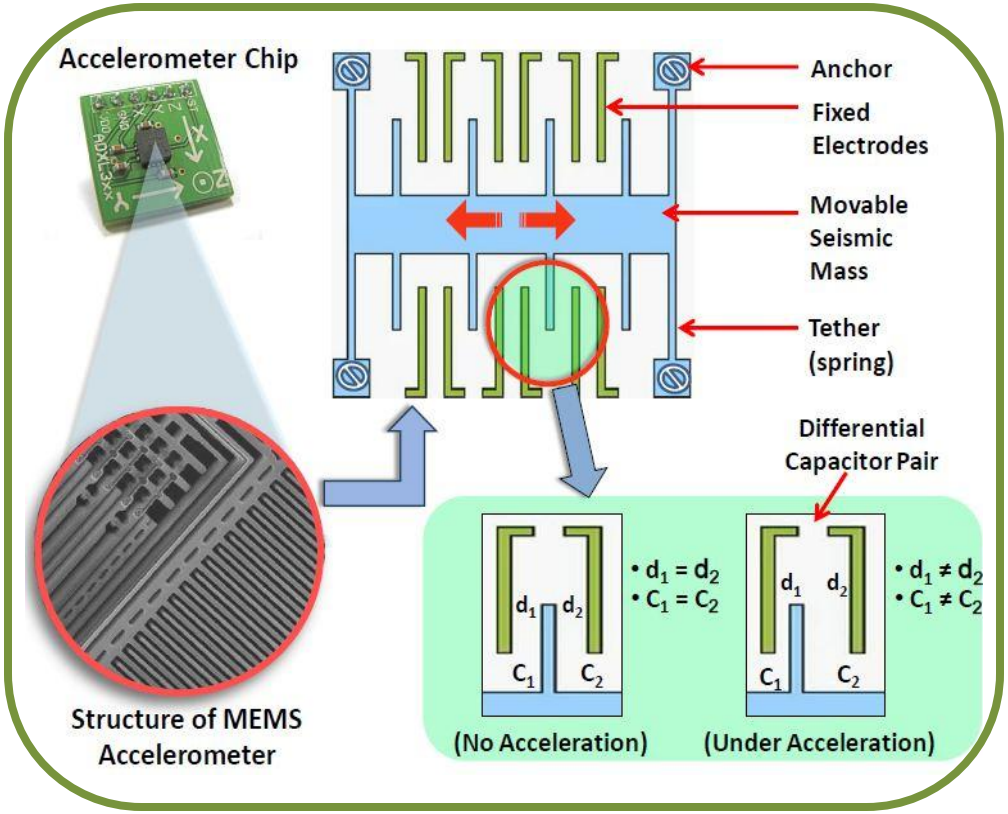


Internal structure of an accelerometer



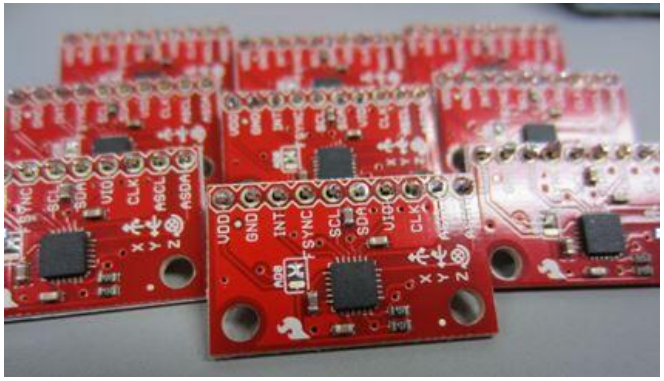
Reasons for difference in accelerometers

- Manufacturing imperfections
- Idiosyncrasies due to QFN and LGA Packaging
- Subtle imperfections do not alter the rated functionality
- Small imperfections can potentially introduce idiosyncrasies in data



Evaluation and External Impact Analysis

Larger Scale Exploration



80 stand-alone accelerometer chips

27 smartphones and tablets

107 stand-alone chips, smartphones and tablets in total

+

36 time domain and frequency domain features

+

Bagged Decision Trees for ensemble learning
(with accelerometer traces)

Feature Selection

Extract 8 time and 10 frequency domain features from S(i) and I(i)

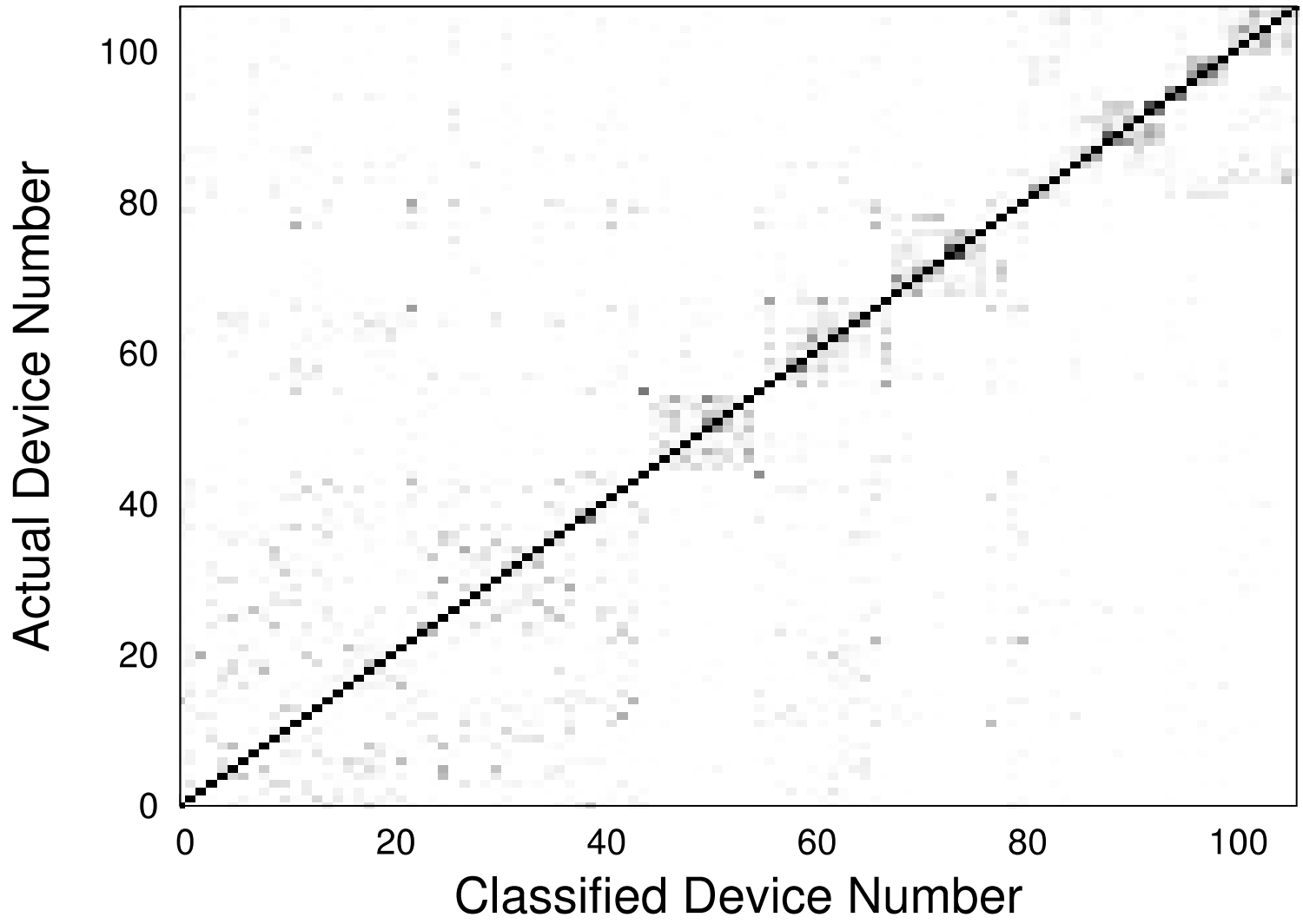
Feature Name	Description
Mean	$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x(i)$
Std-Dev	$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x(i)-\bar{x})^2}$
Average Deviation	$D_{\bar{x}} = \frac{1}{N}\sum_{i=1}^{N}\lvert x(i)-\bar{x}\rvert$
Skewness	$\gamma = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{(x(i)-\bar{x})}{\sigma}\right)^3$
Kurtosis	$\beta = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{(x(i)-\bar{x})}{\sigma}\right)^4 - 3$
RMS Amplitude	$A = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x(i))^2}$
Lowest Value	$L = (Min(x(i))\lvert_{i=1\ to\ N})$
Highest Value	$H = (Max(x(i))\lvert_{i=1\ to\ N})$

Time domain features

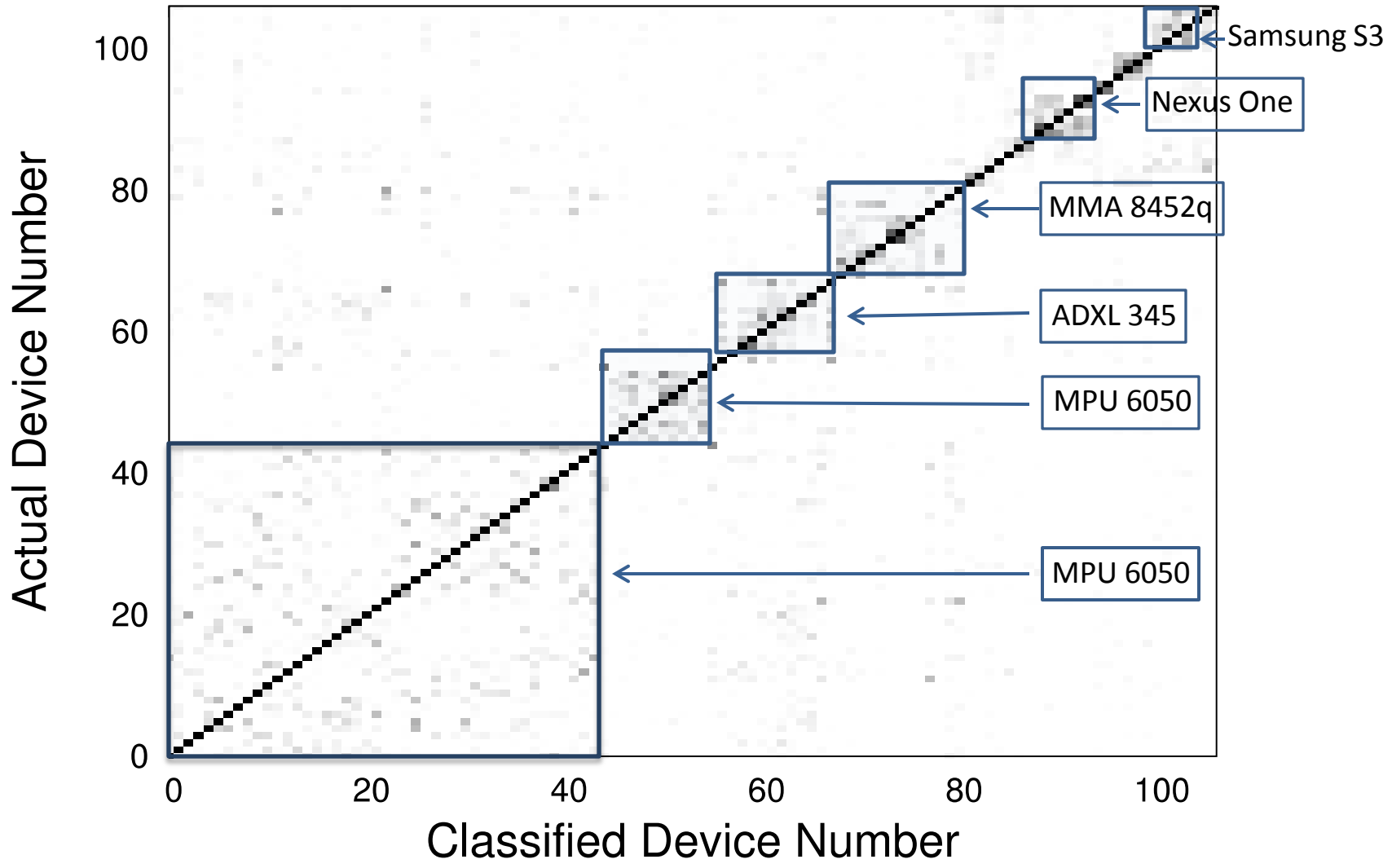
Feature Name	Description
Spec. Std Dev	$\sigma_s = \sqrt{\left(\sum_{i=1}^{N}(y_f(i))^2 * y_m(i)\right) \Big/ \left(\sum_{i=1}^{N} y_m(i)\right)}$
Spec. Centroid	$C_s = \left(\sum_{i=1}^{N} y_f(i) y_m(i)\right) \Big/ \left(\sum_{i=1}^{N} y_m(i)\right)$
Spec. Skewness	$\gamma_s = \left(\sum_{i=1}^{N}(y_m(i)-C_s)^3 * y_m(i)\right) / \sigma_s^3$
Spec. Kurtosis	$\beta_s = \left(\sum_{i=1}^{N}(y_m(i)-C_s)^4 * y_m(i)\right) / \sigma_s^4 - 3$
Spectral Crest	$CR_s = (Max(y_m(i))\lvert_{i=1\ to\ N}) / C_s$
Irregularity-K	$IK_s = \sum_{i=2}^{N-1}\left\lvert y_m(i) - \frac{y_m(i-1)+y_m(i)+y_m(i+1)}{3}\right\rvert$
Irregularity-J	$IJ_s = \frac{\sum_{i=1}^{N-1}(y_m(i)-y_m(i+1))^2}{\sum_{i=1}^{N-1}(y_m(i))^2}$
Smoothness	$S_s = \sum_{i=2}^{N-1}\left\lvert 20.log(y_m(i)) - \frac{\left(20.log(y_m(i-1))+20.log(y_m(i))+20.log(y_m(i+1))\right)}{3}\right\rvert$
Flatness	$F_s = \left(\prod_{i=1}^{N} y_m(i)\right)^{\frac{1}{N}} \Big/ \left(\left(\sum_{i=1}^{N} y_m(i)\right)/N\right)$
Roll Off	$R_s = \frac{SampleRate}{N} * n \Big\lvert \sum_{i=1}^{n} y_m < Threshold$

Frequency domain features

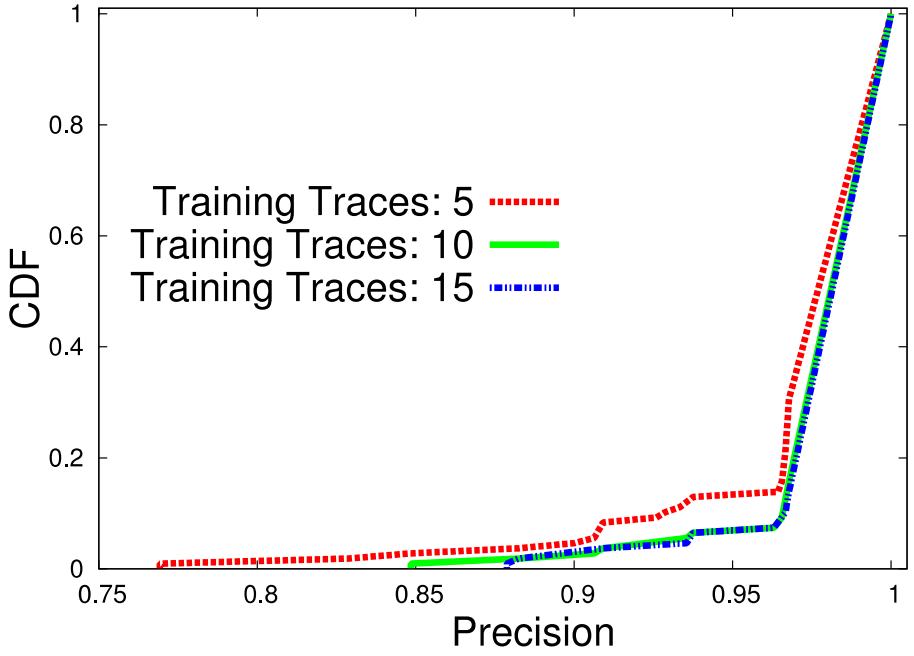
Overall classification performance

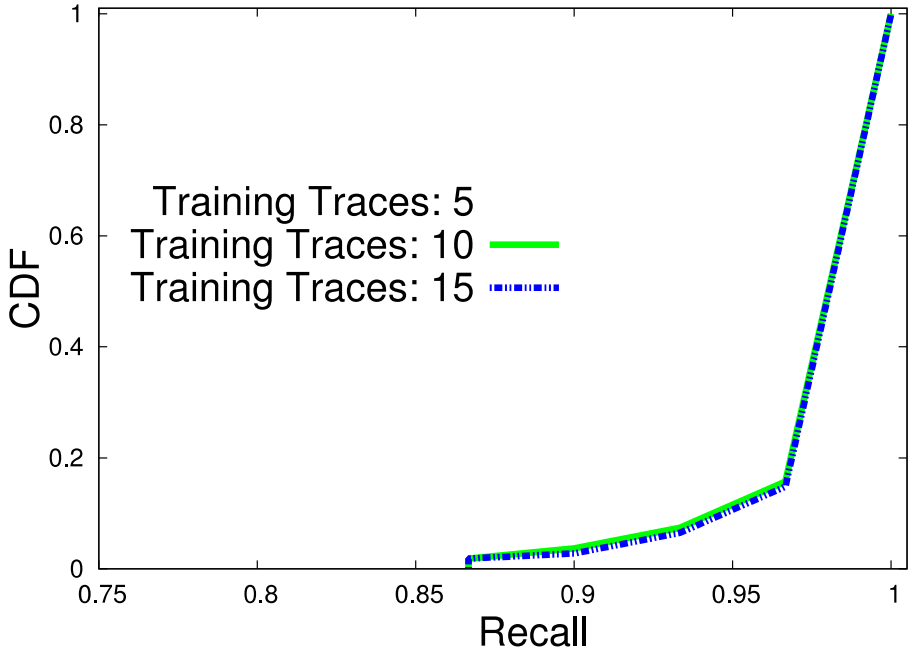


Overall classification performance



Precision and Recall





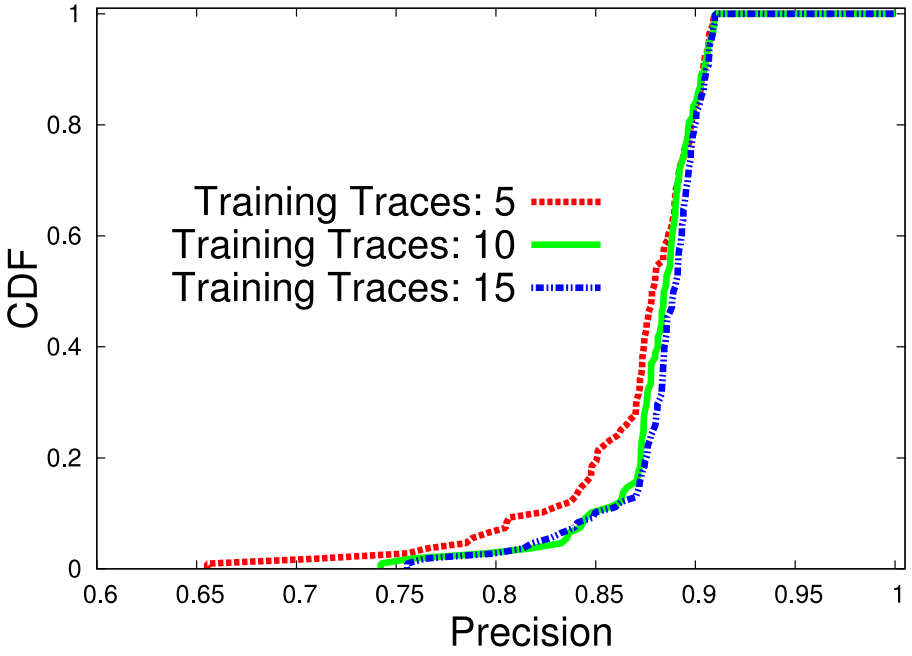
worst case precision & recall > 76%

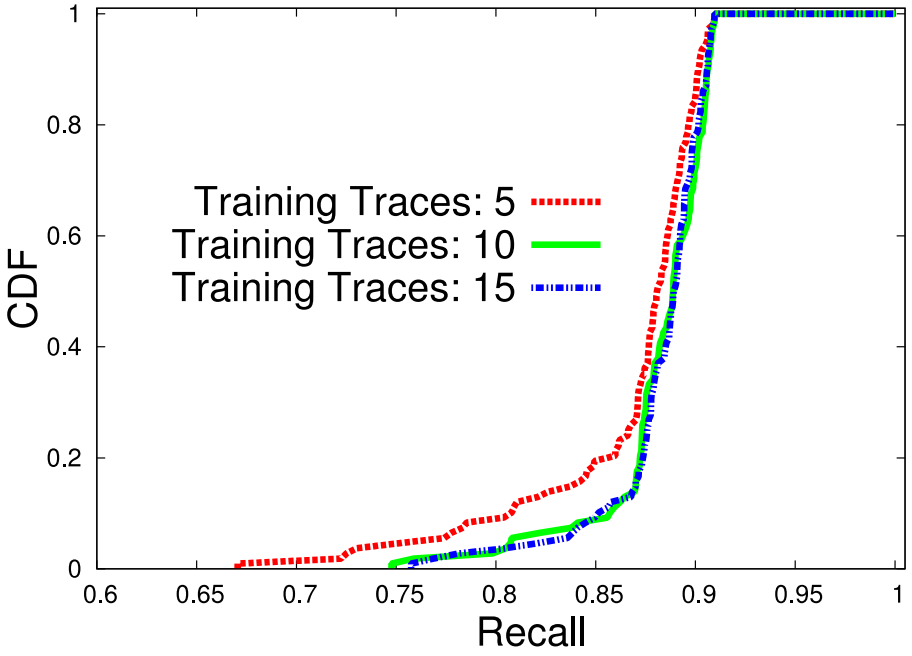
average precision & recall > 99%

Questions

- Is the external vibration mandatory for fingerprinting the accelerometers?
- What is the impact of smartphone CPU load on fingerprints?
- Does the fingerprint manifest only at faster sampling rates?
- Does the system need to be aware of the surface on which device is placed?

Precision and Recall Without Vibration





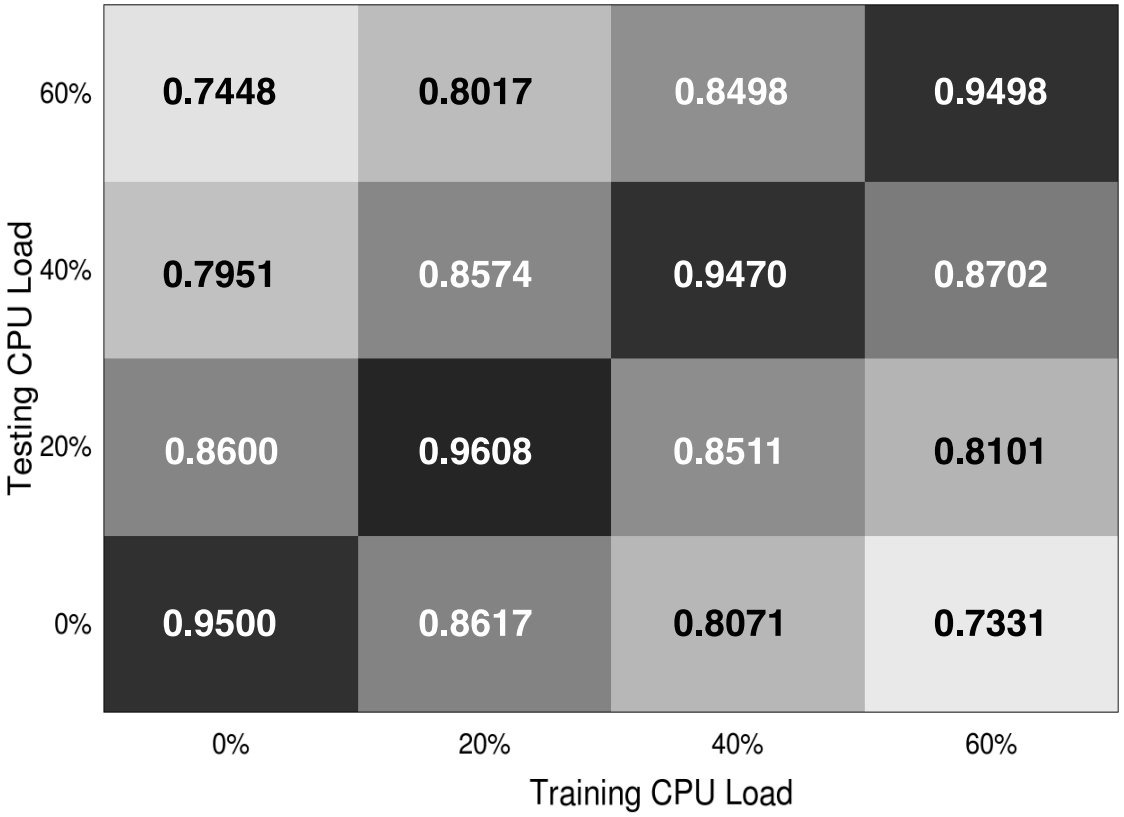
worst case precision & recall > 66%

average precision & recall > 88%

Natural Questions

- Is the external vibration mandatory for fingerprinting the accelerometers?
- What is the impact of smartphone CPU load on fingerprints?
- Does the fingerprint manifest only at faster sampling rates?
- Does the system need to be aware of the surface on which device is placed?

Is the system sensitive to CPU load?

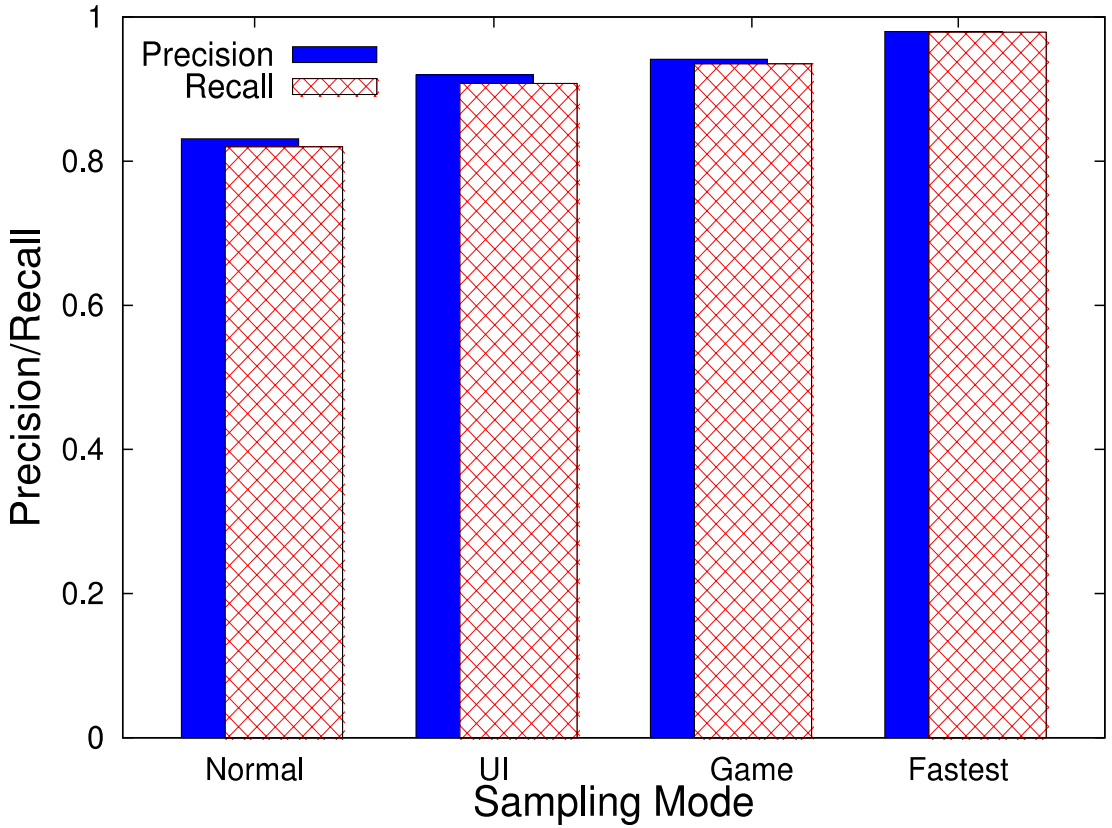


- CPU load matters. But up to 20% difference, high classification precision

Natural Questions

- Is the external vibration mandatory for fingerprinting the accelerometers?
- What is the impact of smartphone CPU load on fingerprints?
- Does the fingerprint manifest only at faster sampling rates?
- Does the system need to be aware of the surface on which device is placed?

Does the fingerprint manifest only at faster sampling rates?

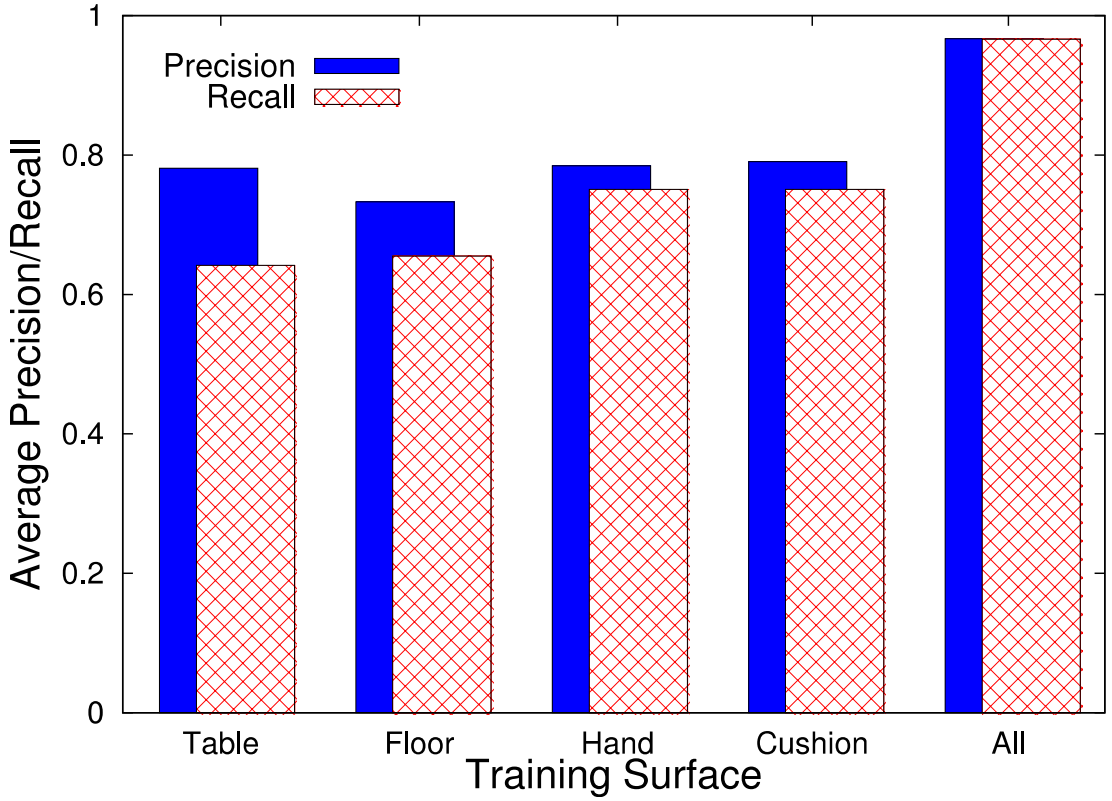


- Even at slower sampling rates, devices exhibit discriminating features
- Likelihood of distinguishing devices improves with faster sampling rates

Natural Questions

- Is the external vibration mandatory for fingerprinting the accelerometers?
- What is the impact of smartphone CPU load on fingerprints?
- Does the fingerprint manifest only at faster sampling rates?
- Does the system need to be aware of the surface on which device is placed?

Does the system need to be aware of the surface on which device is placed?



- Training on different surfaces helps but the system is surface-agnostic

Conclusion and Future Work

- Accelerometers possess fingerprints
- Next step is commercial-grade evaluation
- How to scrub fingerprint from sensor data?

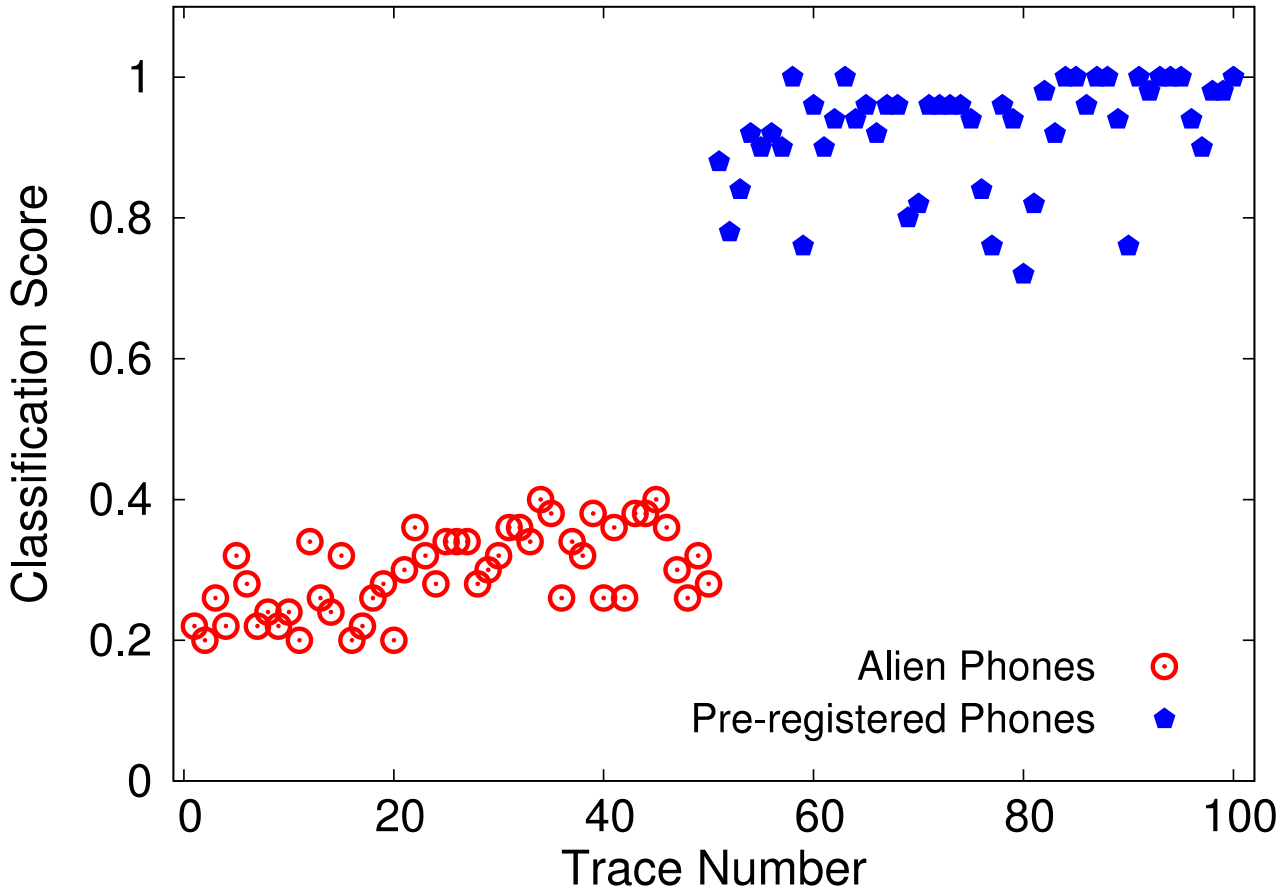
Two objects may be indistinguishable …

… but no two objects are identical

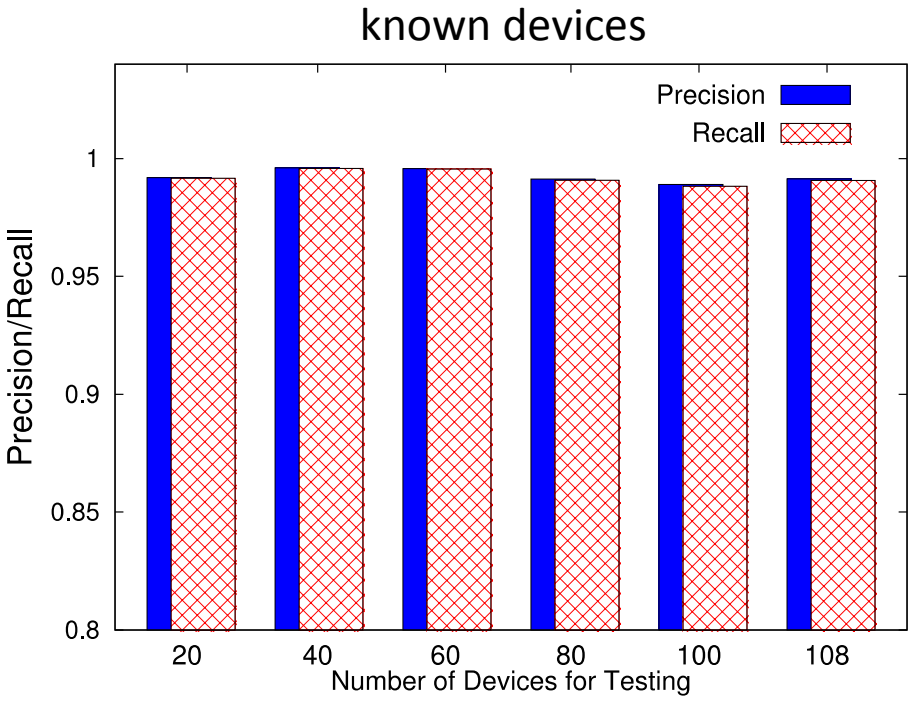
Thank You

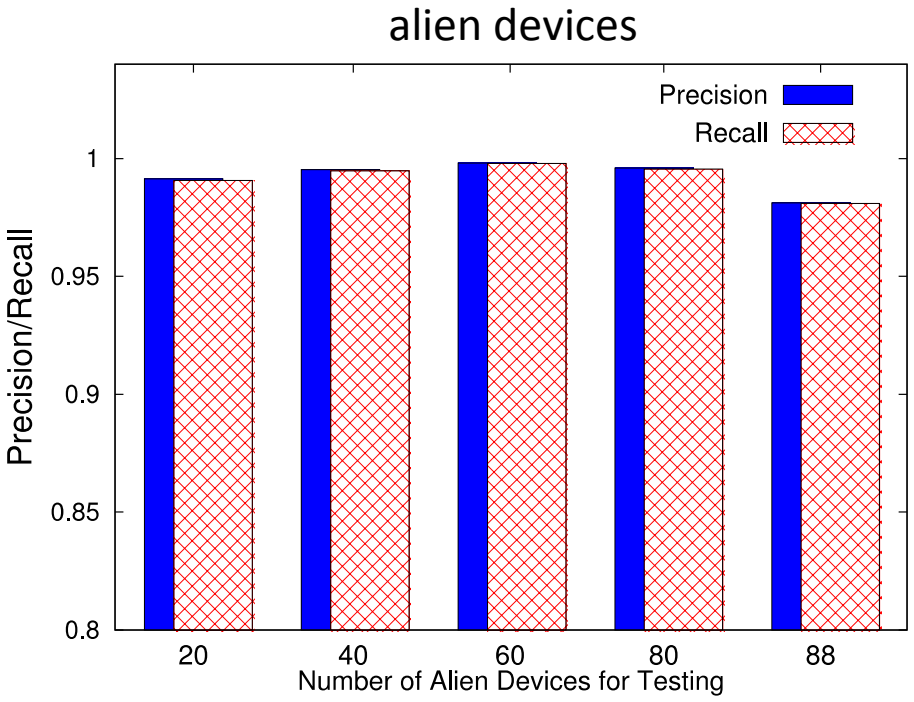
http://web.engr.illinois.edu/~sdey4/

Can we distinguish between an alien phone from a registered phone?



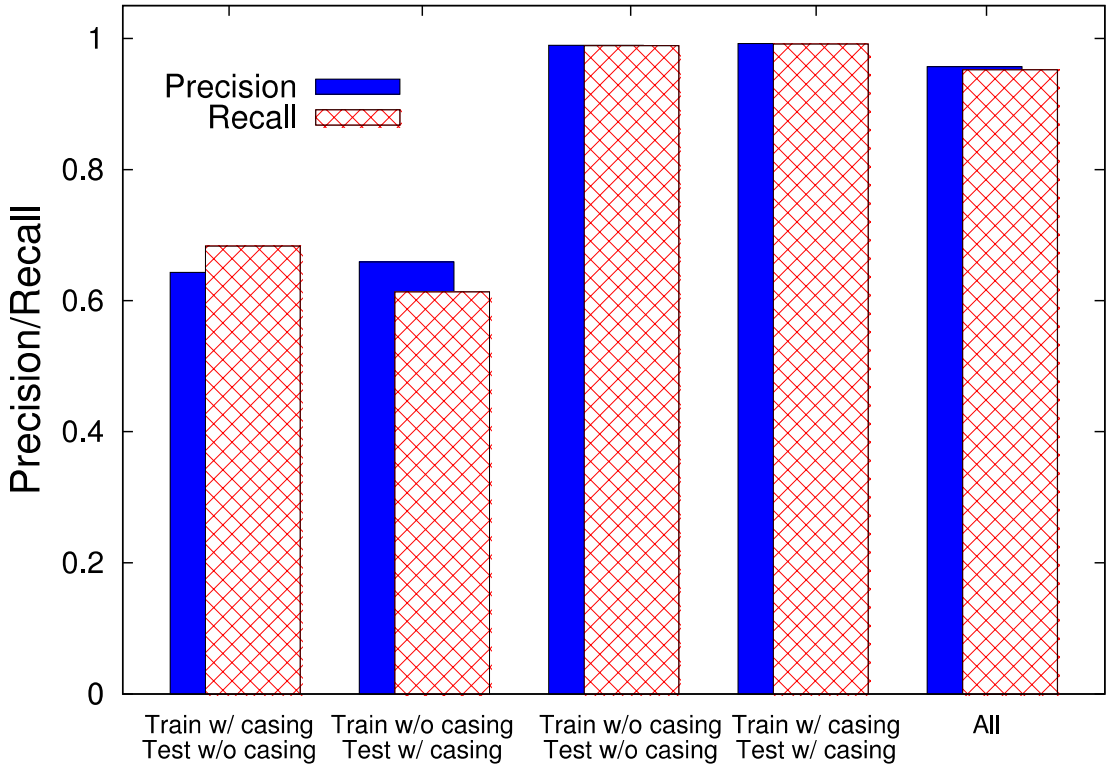
How unique are accelerometer fingerprints?





Even with increasing number of known or alien devices, precision/recall is still high

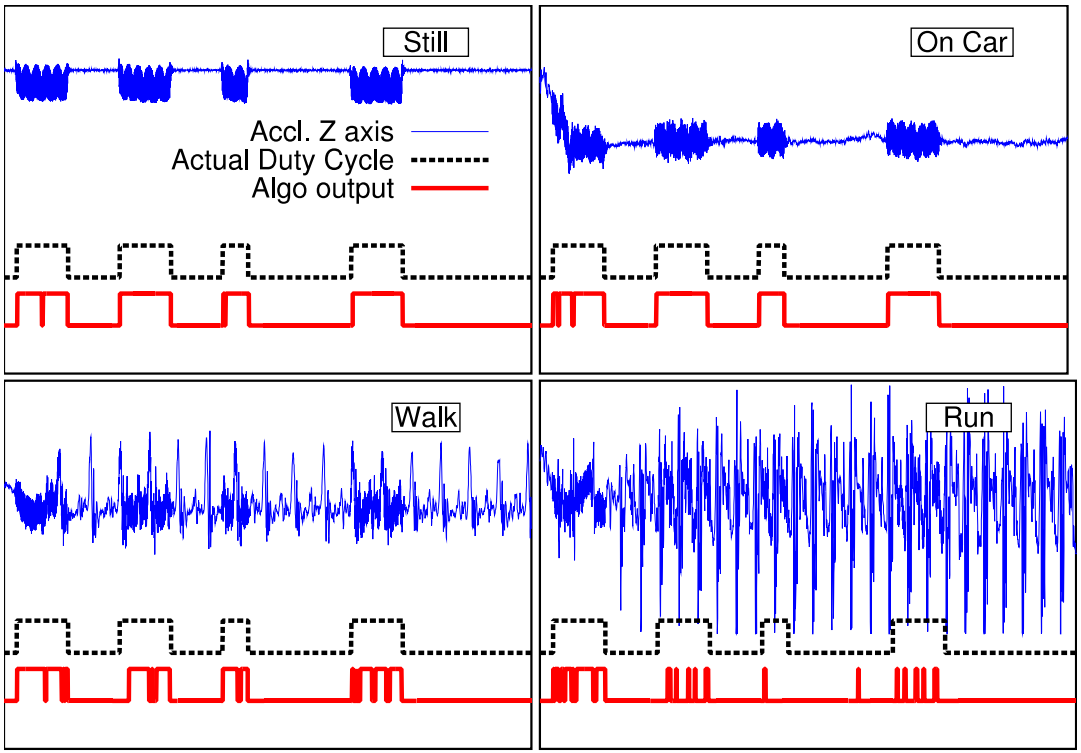
Can we mask a device's fingerprint with a case?



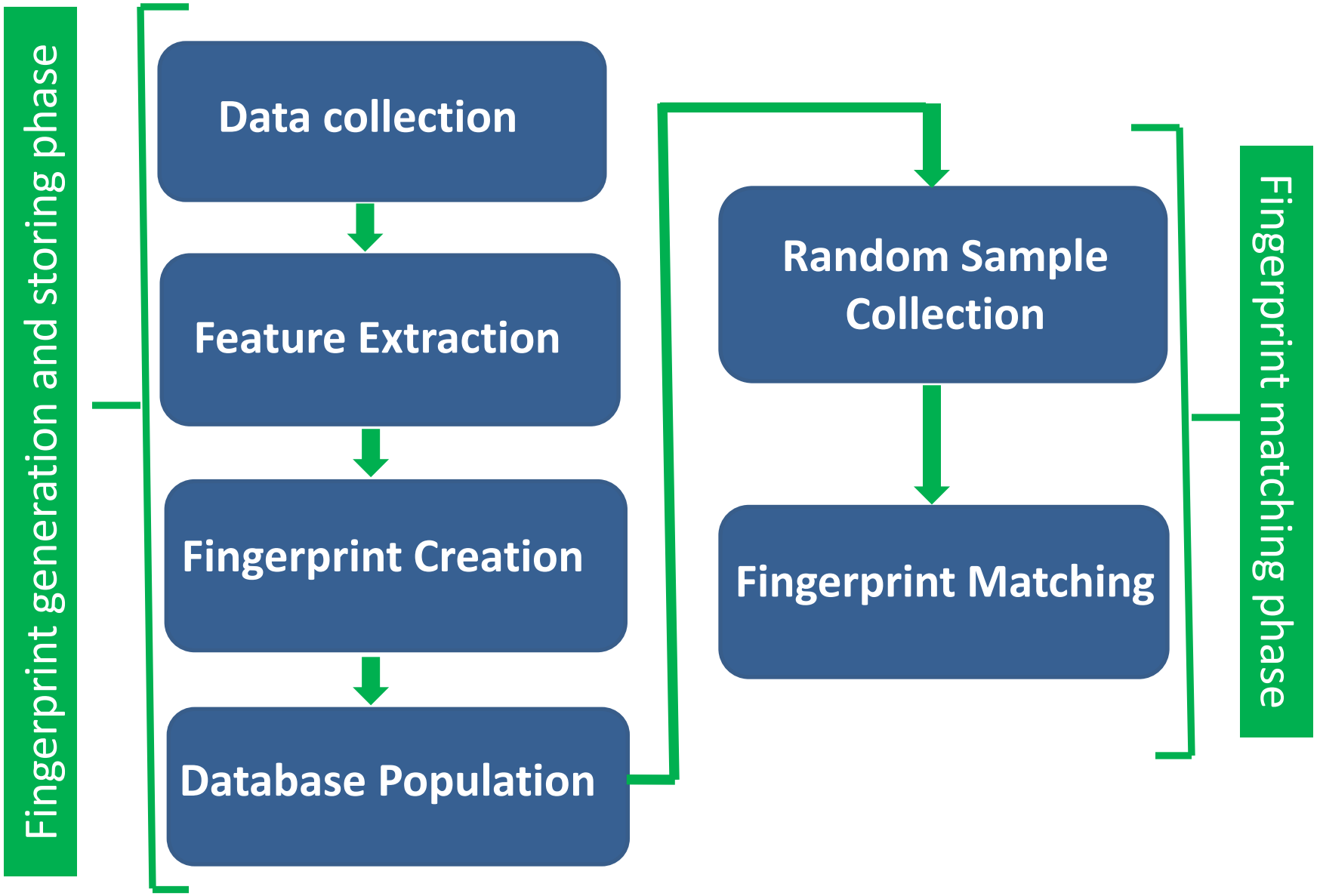
- Accelerometer readings with and without case are different
- Training with and without case still helps classify a device

When to extract a fingerprint in practice?

- Opportunistically under similar conditions
  - e.g. when vibration motor on, CPU load moderate



AccelPrint Design



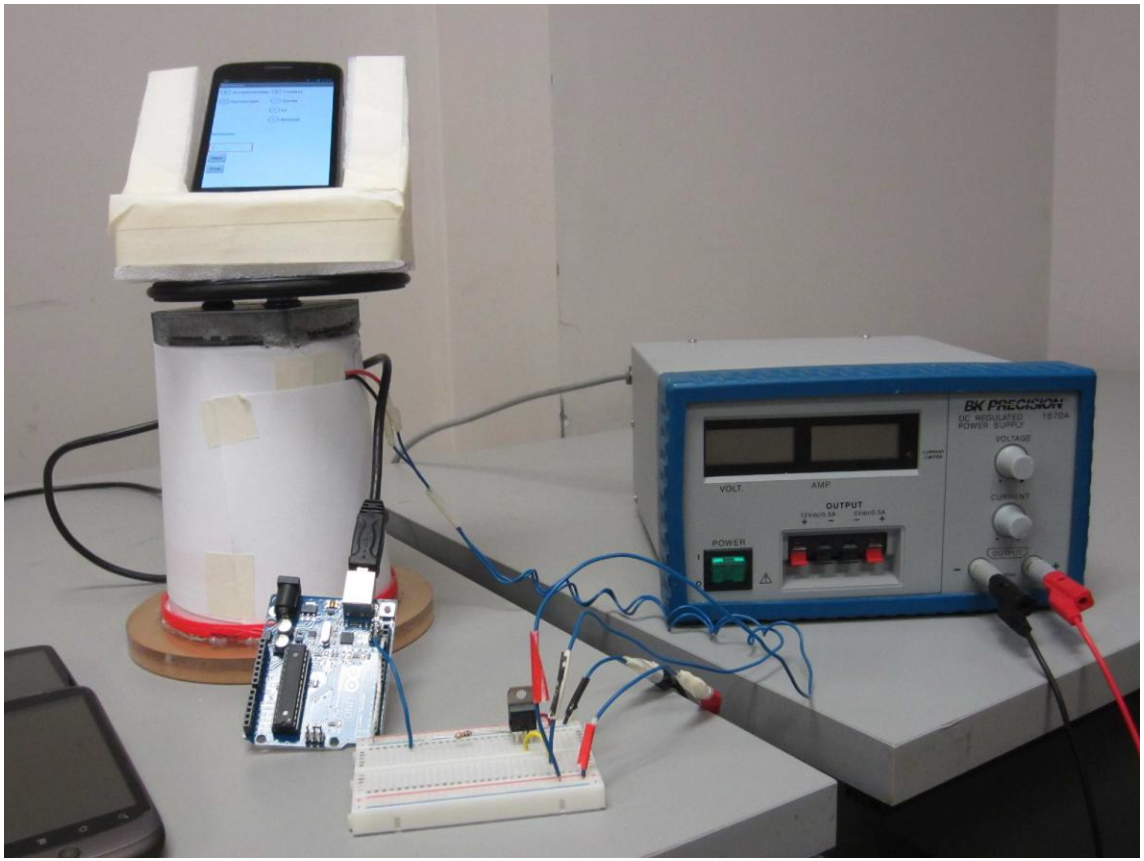
Accelerometer data collection

- Vibrate phone/chip for a certain duration (say 2 sec)
  - Smartphones stimulated with internal vibration motor
- Trace: Accelerometer values during vibration period
  - $\{s_x(i), s_y(i), s_z(i)\}$ be the ith acceleration at time T(i)
    - Root sum square $S(i) = \sqrt{s_x^2(i) + s_y^2(i) + s_z^2(i)}$
  - Samples are not at regular intervals
    - Sampling rate depends on the mode
    - Sampling interval $I(i) = T(i+1) - T(i)$

Fingerprint matching

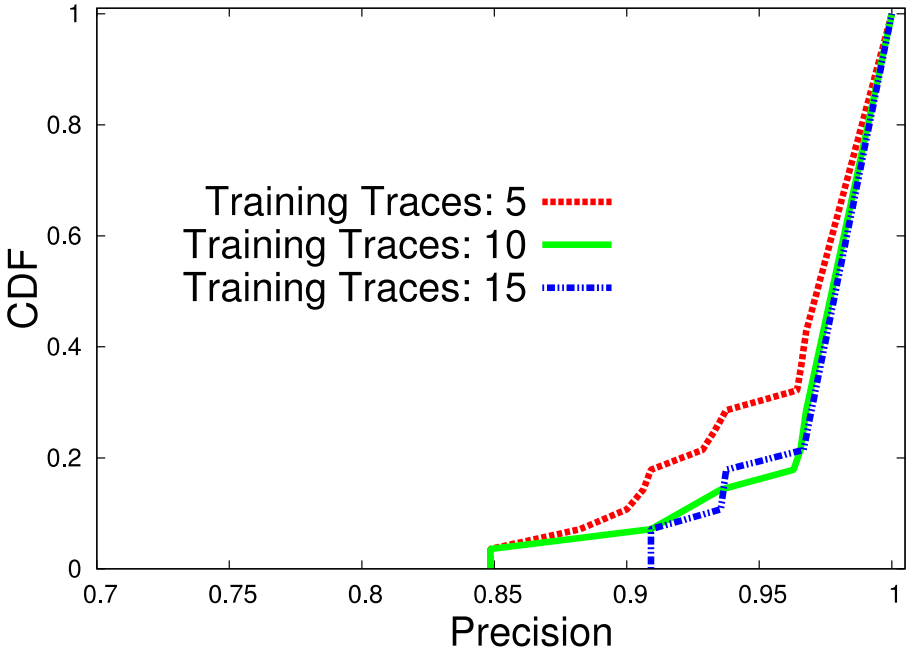
- When a phone is registered
  - AccelPrint is trained with features extracted from multiple (say 10 to 15) traces from that phone
  - Bagged Decision Trees for ensemble learning
- When a phone is tested
  - Extracts features from a single trace
  - Classifier outputs a matching registered phone
    - or “alien” based on classification score

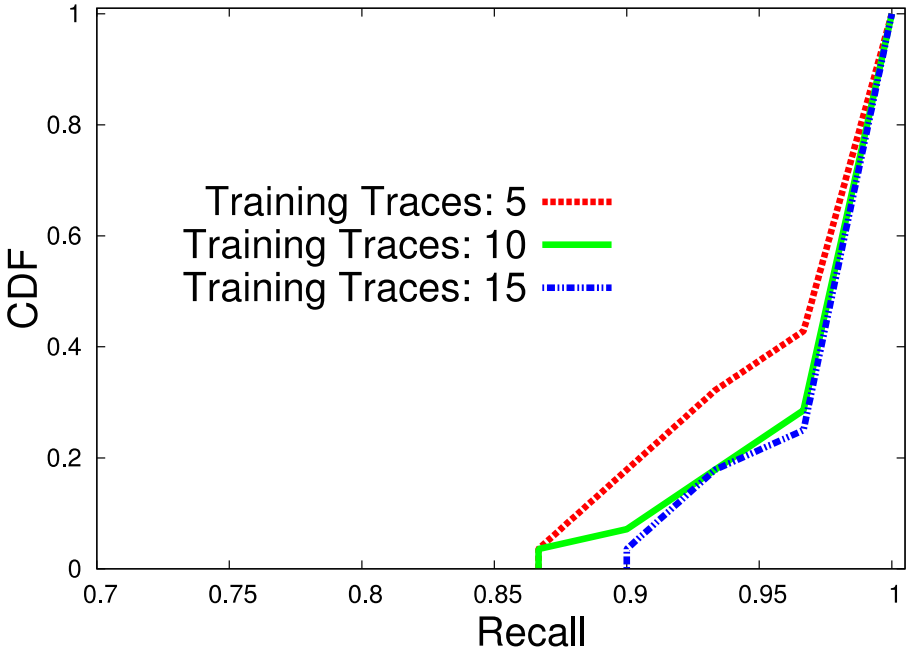
Can we fingerprint a device without vibration?



Rotational setup controlled by Arduino

Can we fingerprint a device without vibration





Even with rotational motion for stimulation, average precision/recall > 97%