SenSig: Practical IoT Sensor Fingerprinting Using Calibration Data

Danté Gray, Maryam Mehrnezhad, Rishad Shafik

Newcastle University, UK

RESEARCH MOTIVATION

WHAT AM I DOING AND WHY?

RESEARCH GOAL

- Develop a solution for augmenting security in an IoT environment
- Fingerprint motion sensors (gyroscopes) using their output alone
  - Output Target : Runtime Calibration Data

RESEARCH MOTIVATION

- Cryptographic/security tools which **factor in IoT constraints** (open research area)
  - Limited Storage
  - Limited Processing Capabilities
  - Limited Power
- The data sensors transmit can have physical, **real-world consequences**
  - IIoT (Temperature/Radiation/Proximity Sensor)
  - Dams (Open/close gates based off of readings)
  - Smart Home
- Make a case for the importance of **standardisation** of sensors in IoT
  - What sensor data can be made available
  - How sensor data can be accessed

SUMMARY

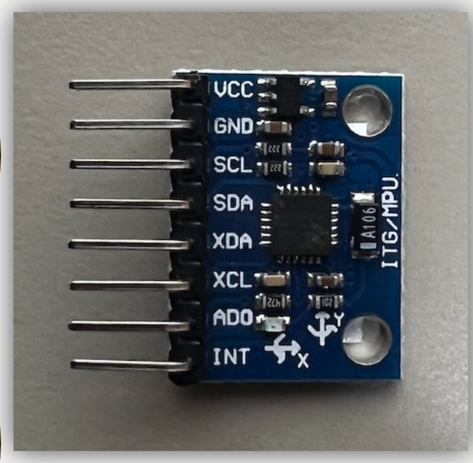
- **Goal:** Develop a solution which augments security in an IoT environment via fingerprinting.
- **Requirements:** Computationally light & does not require usage/storage of external data.
- **Use Case**: Identification

EQUIPMENT

TOOLS OF THE TRADE

HARDWARE

- Arduino Mega (Microcontroller)
- Inertial Measurement Units:
  - MPU-6050 (primary)
  - ICM-20948



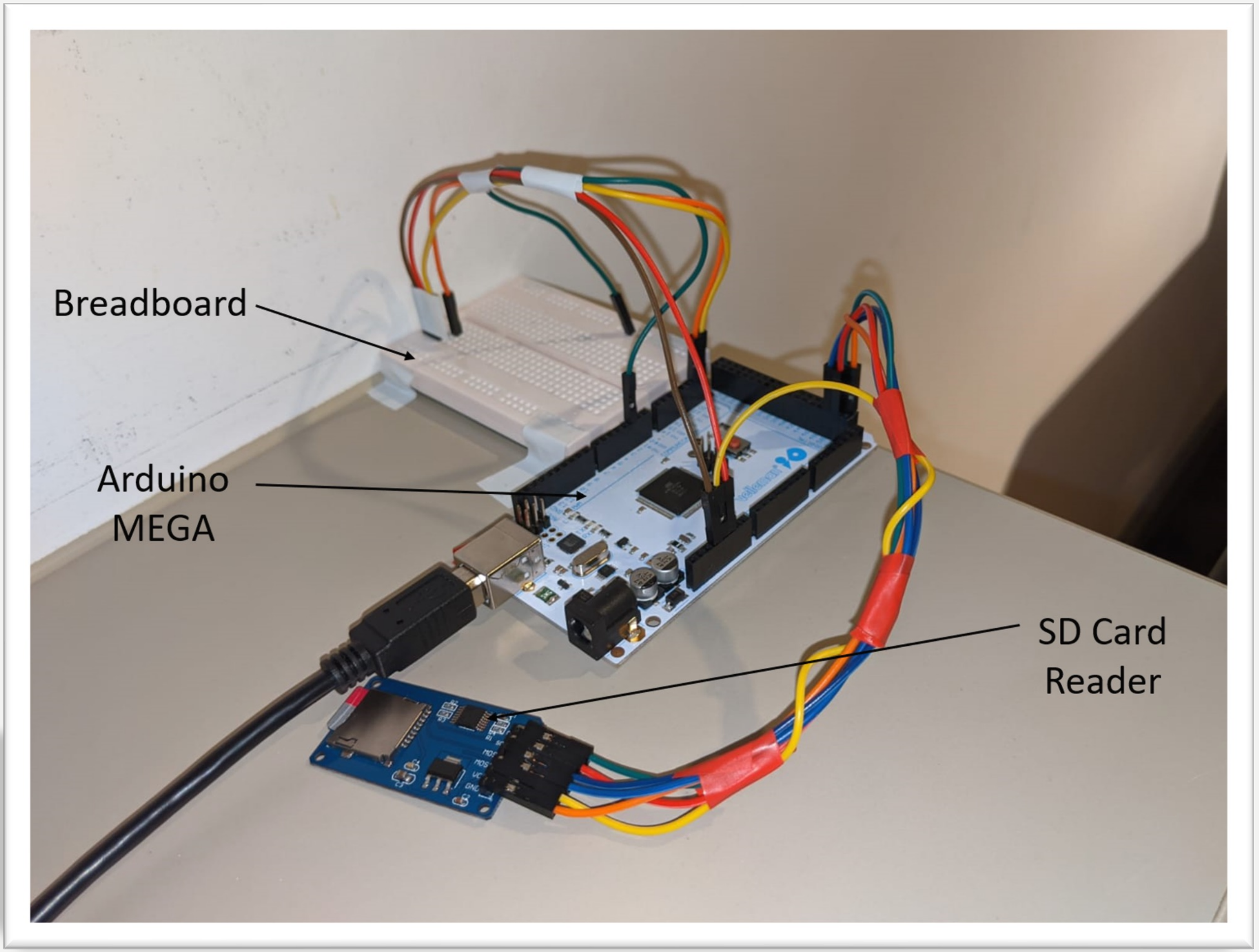


Image of one of the data collection environment

SOFTWARE

- Arduino IDE
  - IMU Communication
- MATLAB
  - Signal Processing
  - Proof-of-Concept
- Various Python Scripts

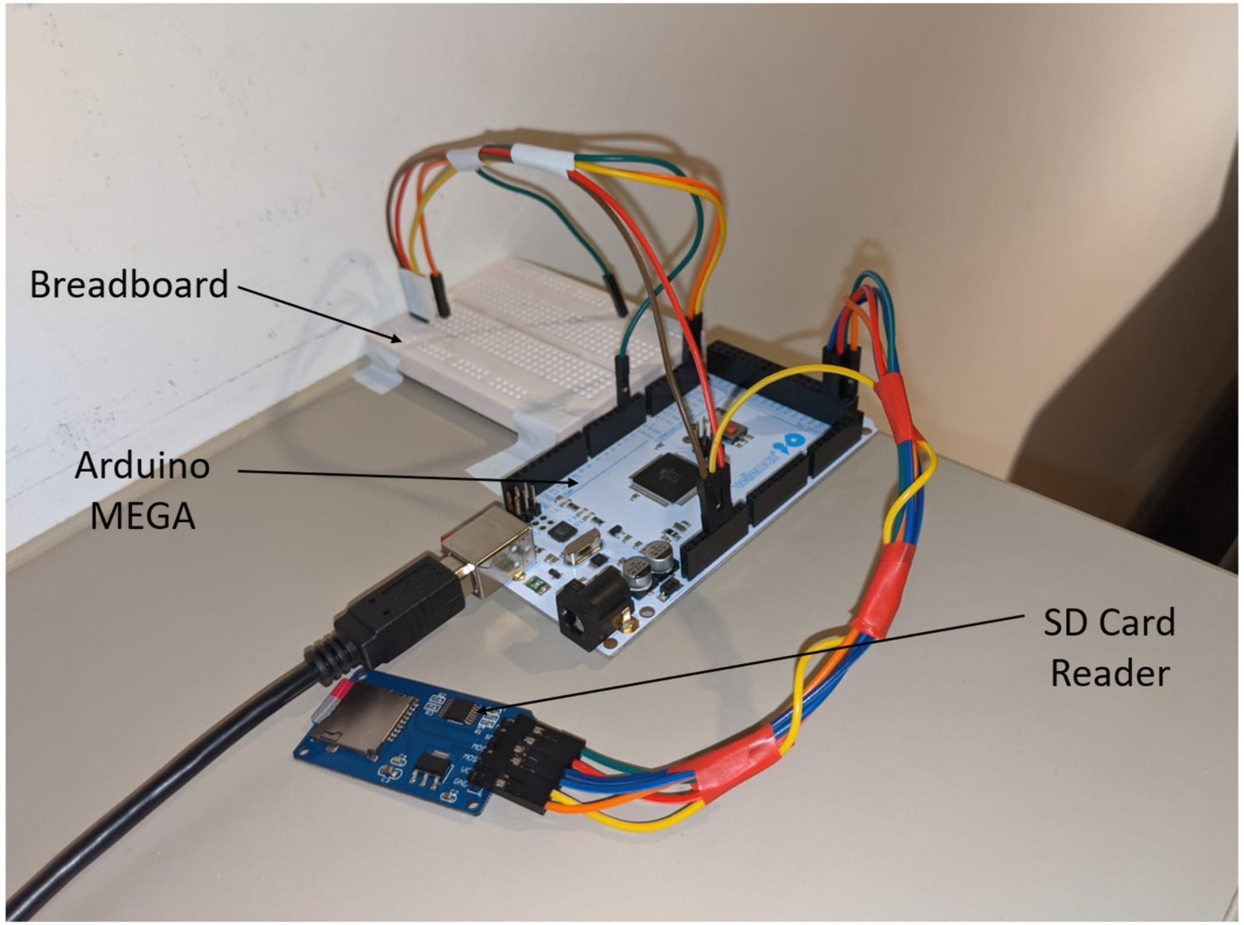


Image of one of the data collection environment

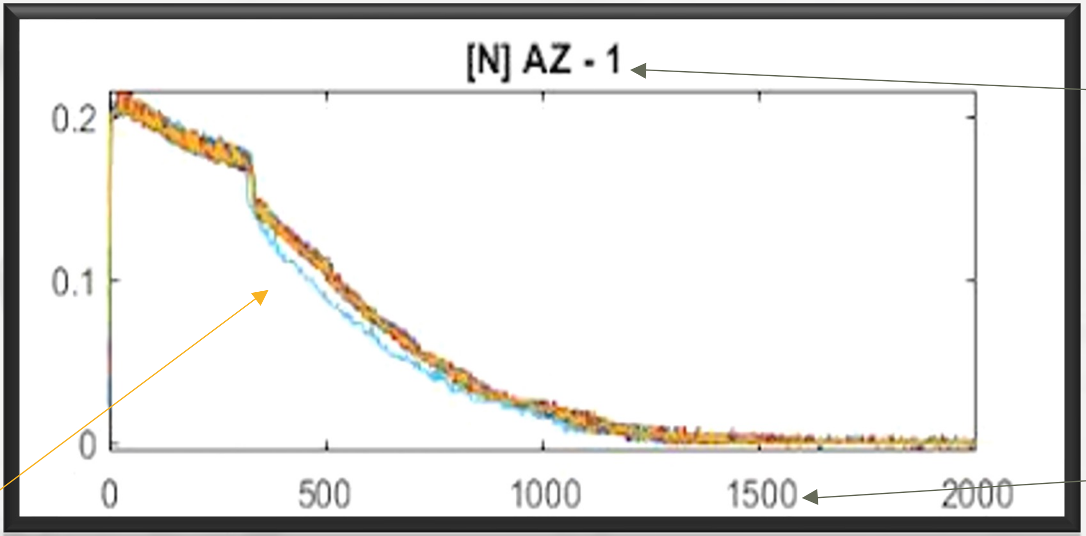
DATASET GENERATION

THERE ARE TWO TYPES OF PEOPLE, THOSE THAT CAN EXTRAPOLATE FROM AN INCOMPLETE DATA SET

DATASET GENERATION

Dataset Size: 23 sensors (mpu-6050)

Rounds of Data per Sensor: 31 rounds (runtime calibration data)



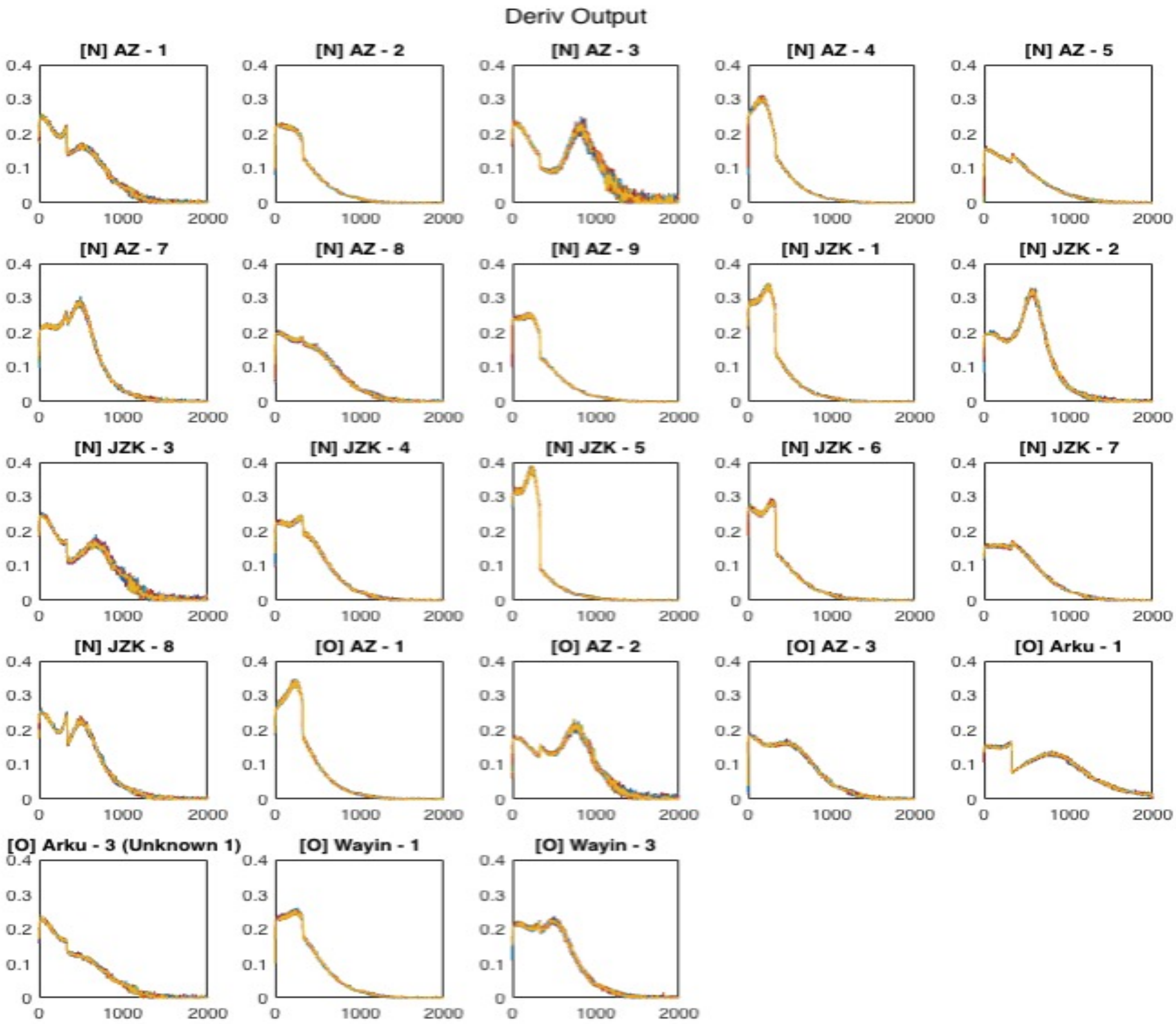
Device Name

X^{th} Datapoint

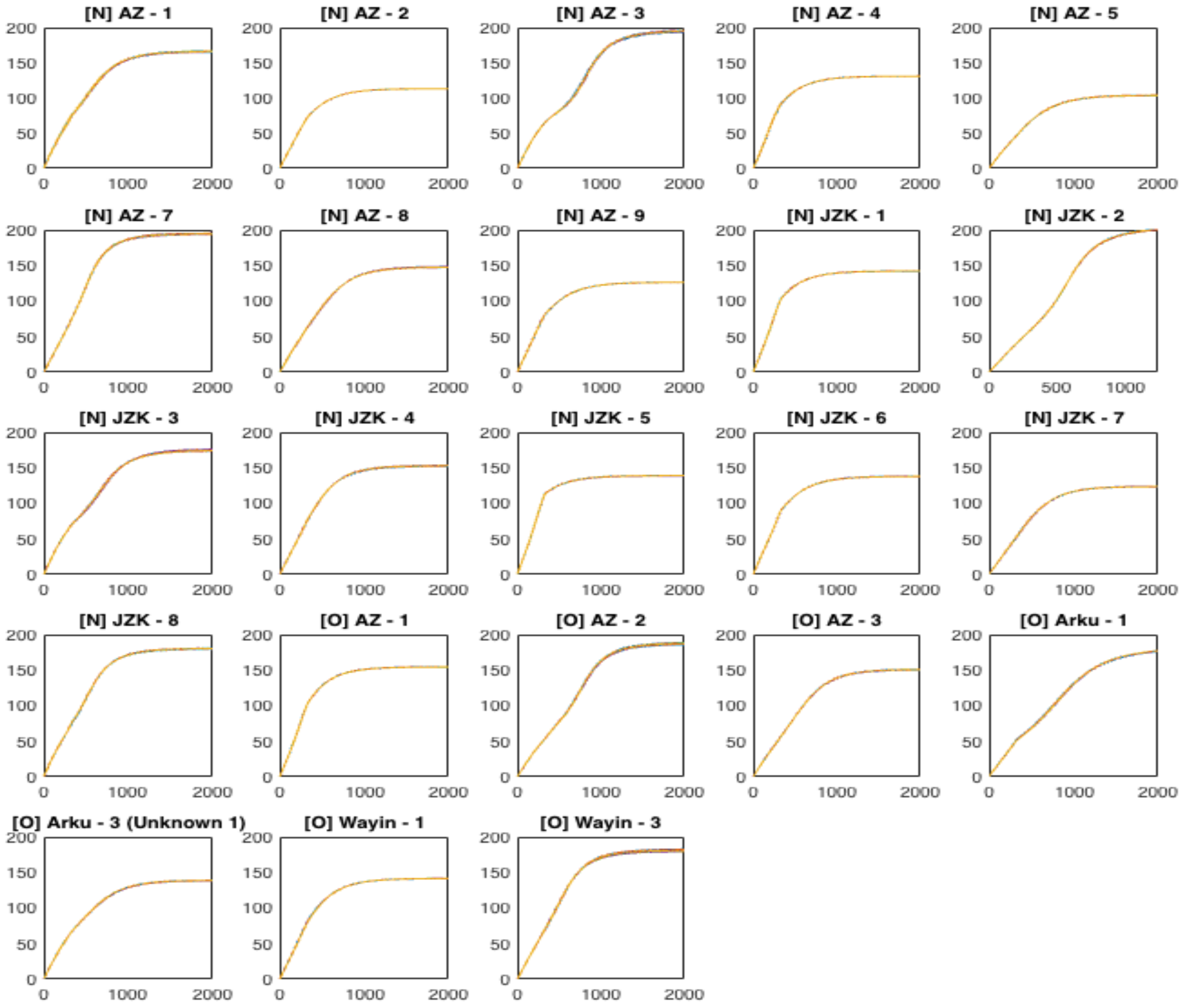
Single Instance of
Runtime Calibration Data (per line)
[1^{st} Derivative]

FULL DATASET (1ST DERIVATIVE)

MPU-6050



FULL DATASET (GYRO DATA)
MPU-6050



SIGNAL PROCESSING

01001001 01100110 00100000 01111001 01101111 01110101 00100000 01100011 01100001 01101110 00100000 01110010
01100101 01100001 01100100 00100000 01110100 01101000 01101001 01110011 00100000 01111001 01101111 01110101
00100000 01101110 01100101 01100101 01100100 00100000 01110100 01101111 00100000 01100111 01101111 00100000
01101111 01110101 01110100 01110011 01101001 01100100 01100101 00100000 01101101 01101111 01110010 01100101

SIGNAL PROCESSING

The signal processing for this research is broken down into three stages:

- Pre-Processing
- Quantisation
- Template Generation

PRE-PROCESSING

YES, WE HAVE TO PROCESS BEFORE WE CAN PROCESS.
WHAT A CRAZY WORLD WE LIVE IN…

PRE-PROCESSING

- Noise Removal
- Data Conversion
- Data Truncation

NOISE REMOVAL {1}

Primary focus of this stage is to eliminate the unwanted **noise** was present after data collection.

This noise was present in the form of **spikes**.

This is addressed using ***Median Filtering***.

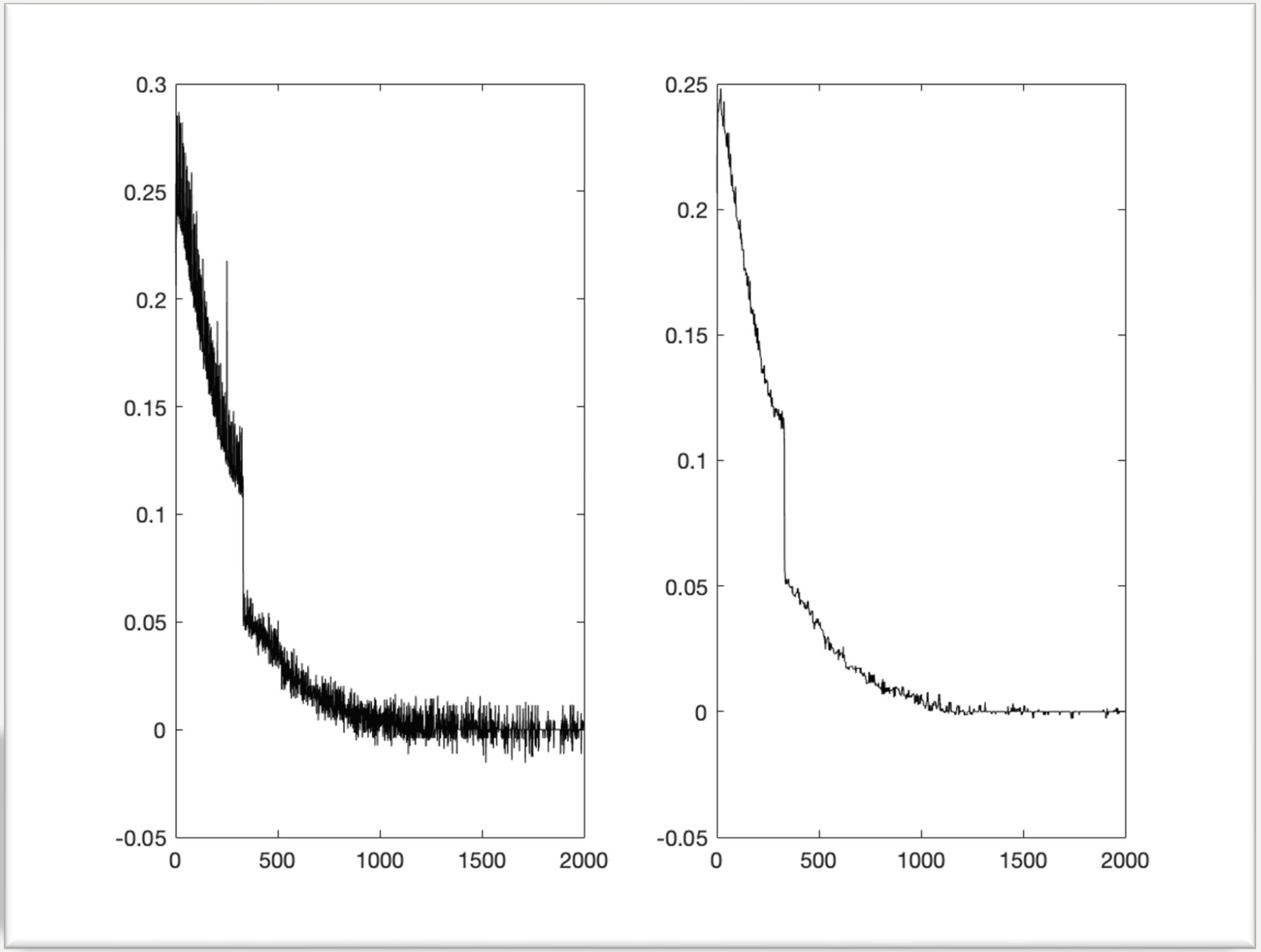
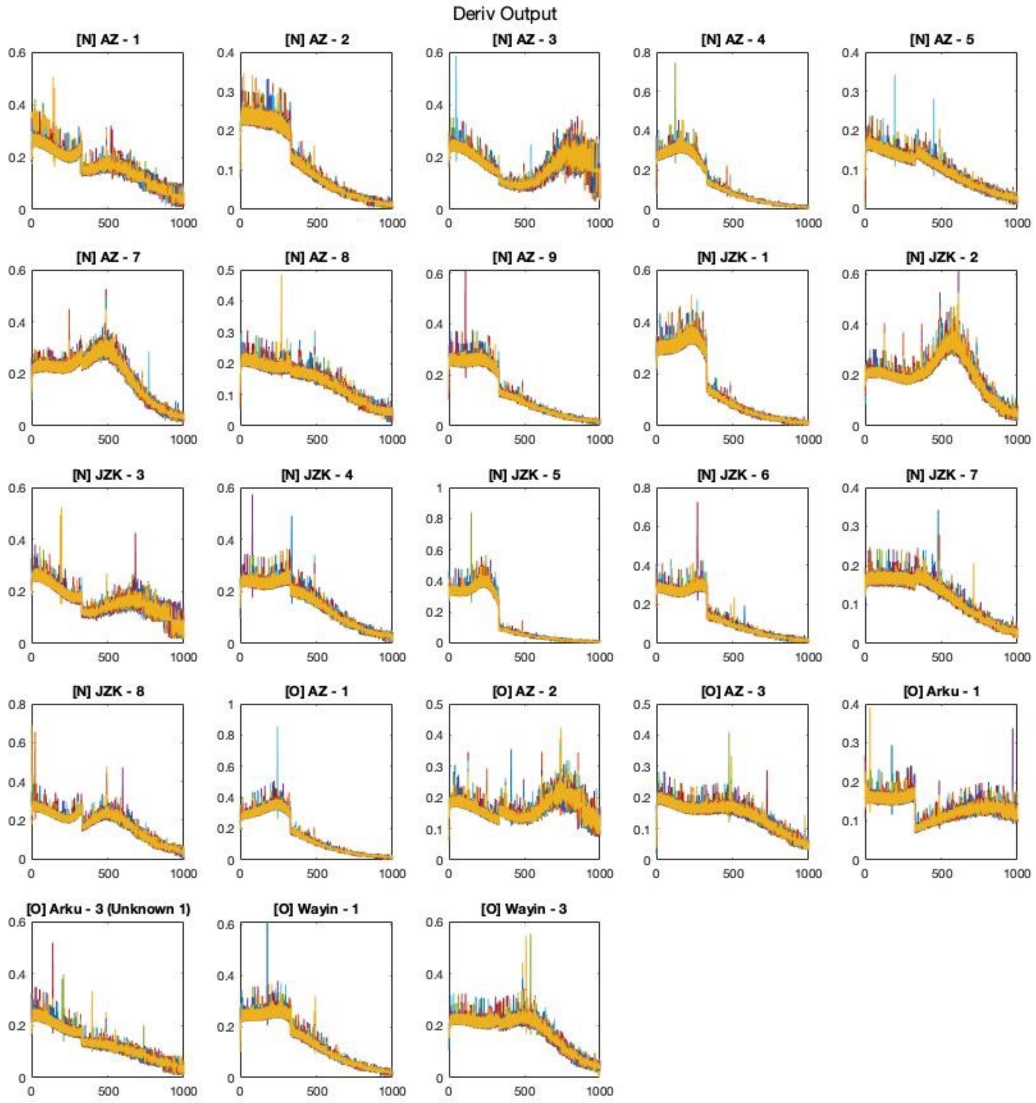
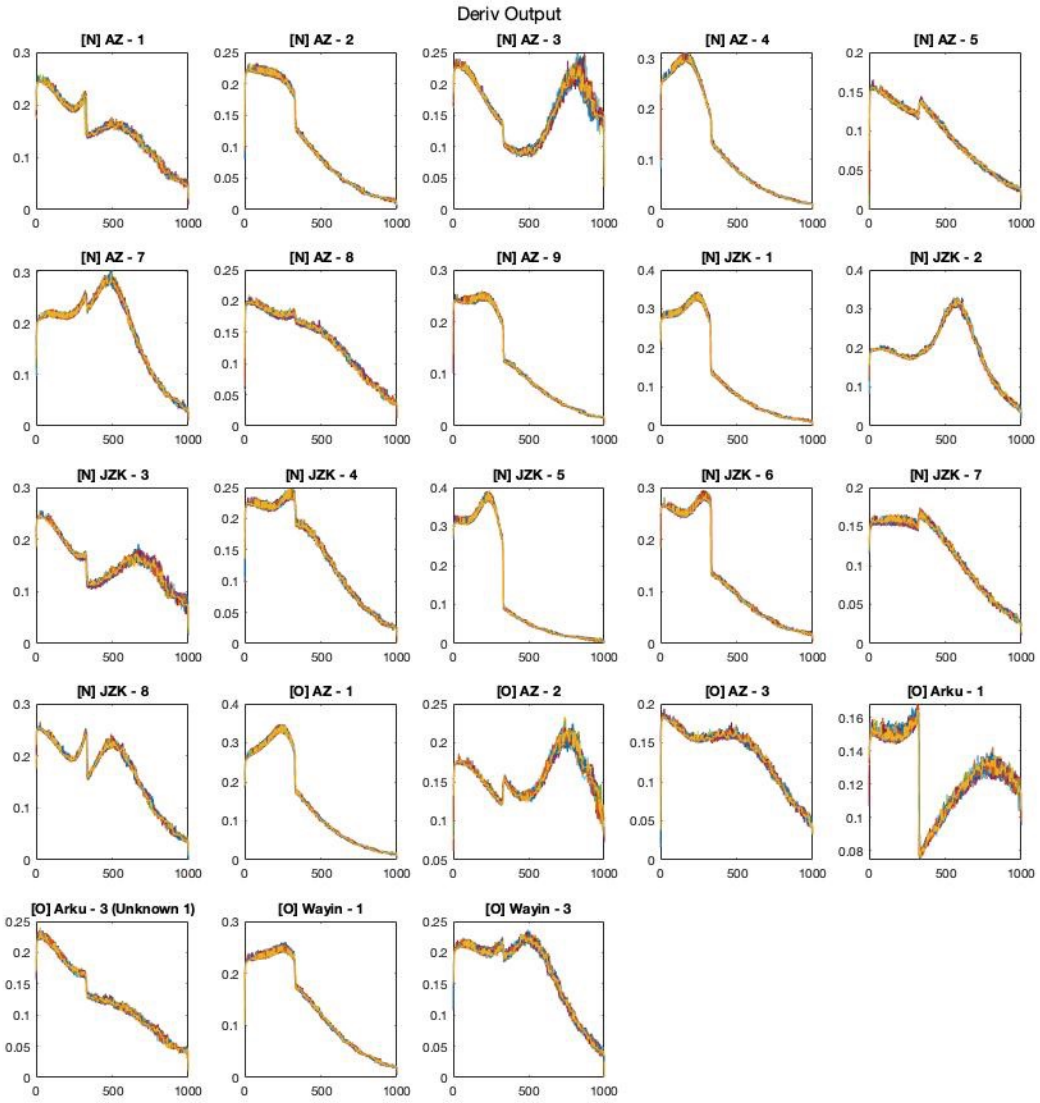


Image of signal pre/post noise removal


Deriv Output
[N] AZ - 1
[N] AZ - 2
[N] AZ - 3
[N] AZ - 4
[N] AZ - 5
[N] AZ - 7
[N] AZ - 8
[N] AZ - 9
[N] JZK - 1
[N] JZK - 2
[N] JZK - 3
[N] JZK - 4
[N] JZK - 5
[N] JZK - 6
[N] JZK - 7
[N] JZK - 8
[O] AZ - 1
[O] AZ - 2
[O] AZ - 3
[O] Arku - 1
[O] Arku - 3 (Unknown 1)
[O] Wayin - 1
[O] Wayin - 3


Deriv Output
[N] AZ - 1
[N] AZ - 2
[N] AZ - 3
[N] AZ - 4
[N] AZ - 5
[N] AZ - 7
[N] AZ - 8
[N] AZ - 9
[N] JZK - 1
[N] JZK - 2
[N] JZK - 3
[N] JZK - 4
[N] JZK - 5
[N] JZK - 6
[N] JZK - 7
[N] JZK - 8
[O] AZ - 1
[O] AZ - 2
[O] AZ - 3
[O] Arku - 1
[O] Arku - 3 (Unknown 1)
[O] Wayin - 1
[O] Wayin - 3

DATA CONVERSION {2}

Data Format #1: *gyr*

Upon collection, raw gyroscope data is separated into three axes.

We combine this data into one axis through computing the vector length (common practice).

$$gyr_i = \sqrt{gyr_x^2 i + gyr_y^2 i + gyr_z^2 i}$$

Data Format #2: *deriv*

We also compute the **first derivative** of the sensor sequences.

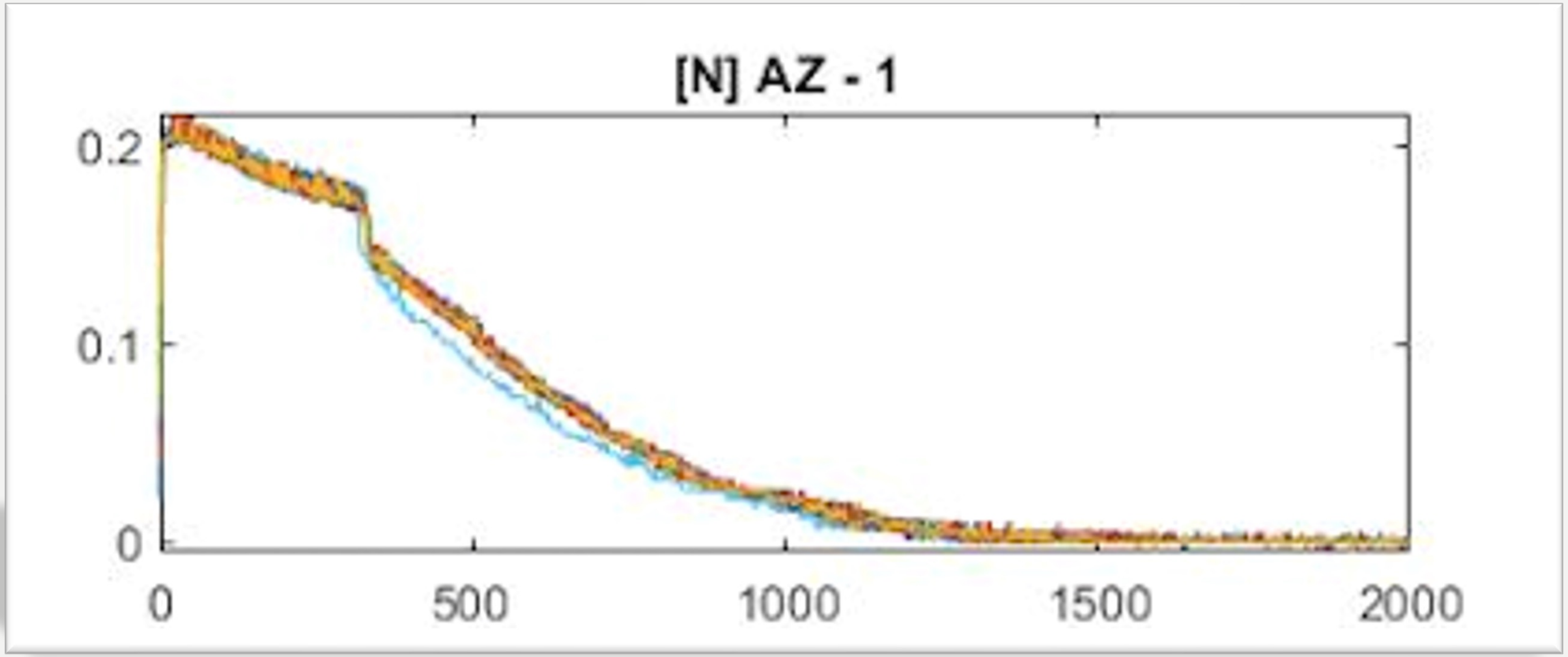
The first derivative was useful for data visualisation.

$$deriv_i = \frac{(gyr_i - gyr_{i-1}) + ((gyr_{i+1} - gyr_{i-1})/2)}{2}$$

TRUNCATION {3}

The sensors take appx. 2000 datapoints to self-calibrate at **start-up**.

Truncation attempts to answer the following: *How much of this data is **fingerprintable**?*



PRE-PROCESSING SUMMARY

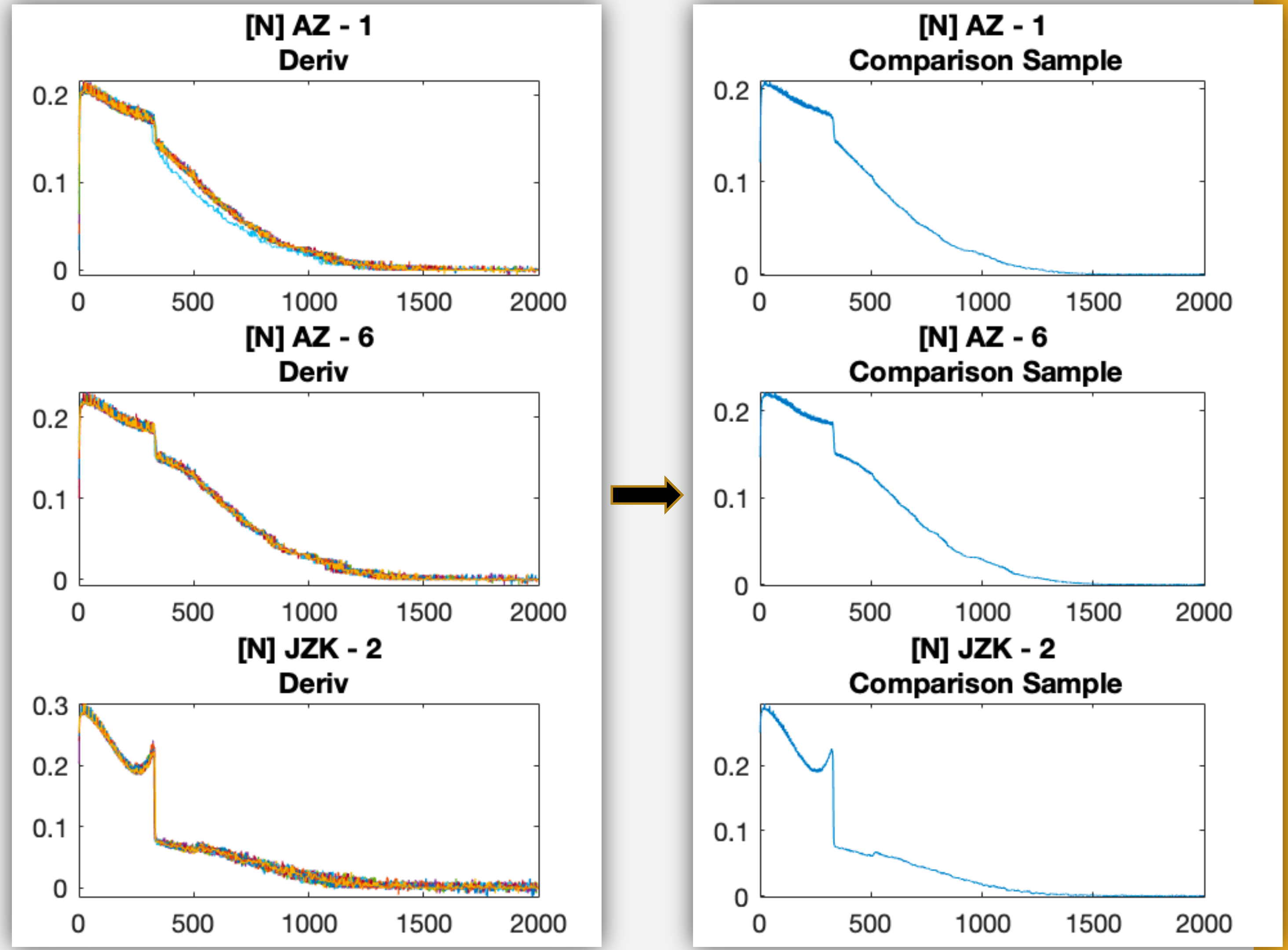
- Noise Removal
- Converting into usable data format
- Data Truncation

TEMPLATE GENERATION

TEMPLATES

In the context of an identification system, **templates** are needed as a **reference** to an entity upon identification.

The controllable variable for template generation is the **number of rounds** of data used.



QUANTISATION

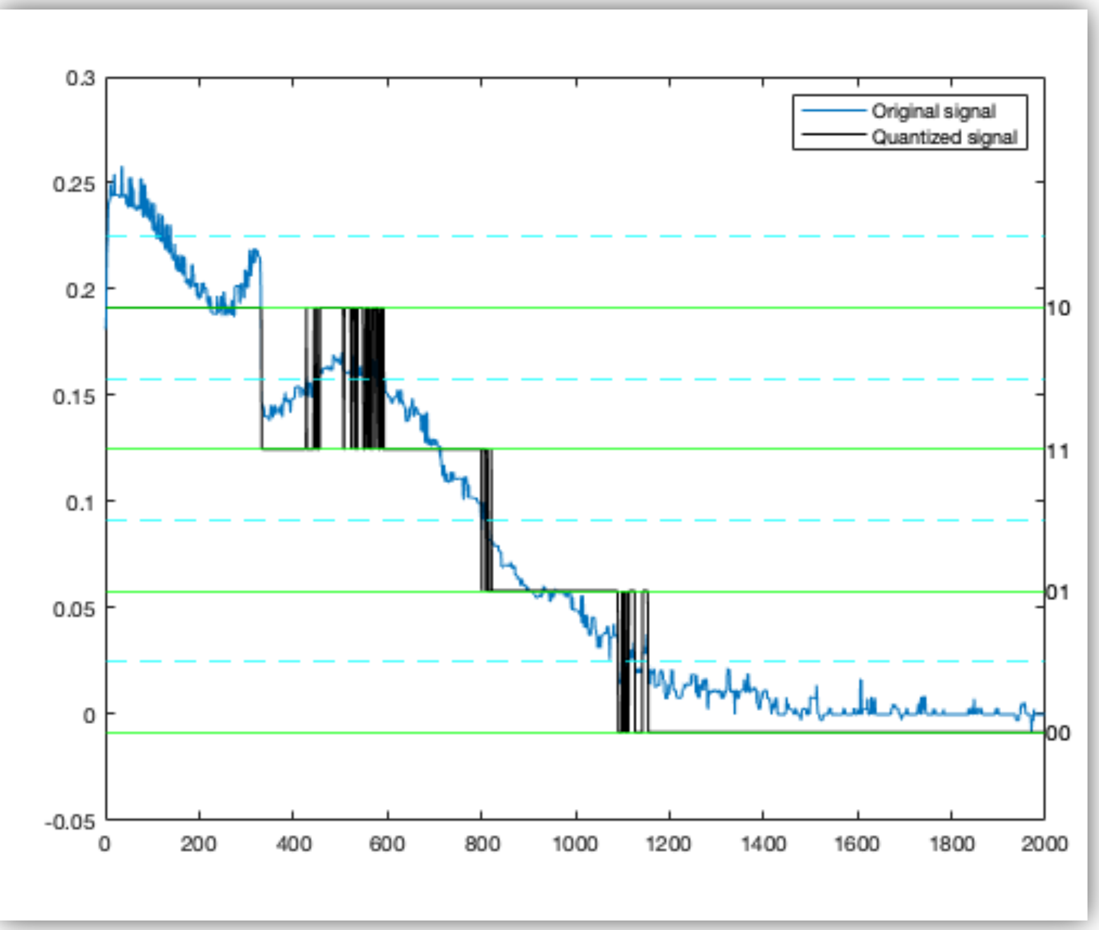
TO BE OR NOT 0010 1011

QUANTISATION EXPLAINED

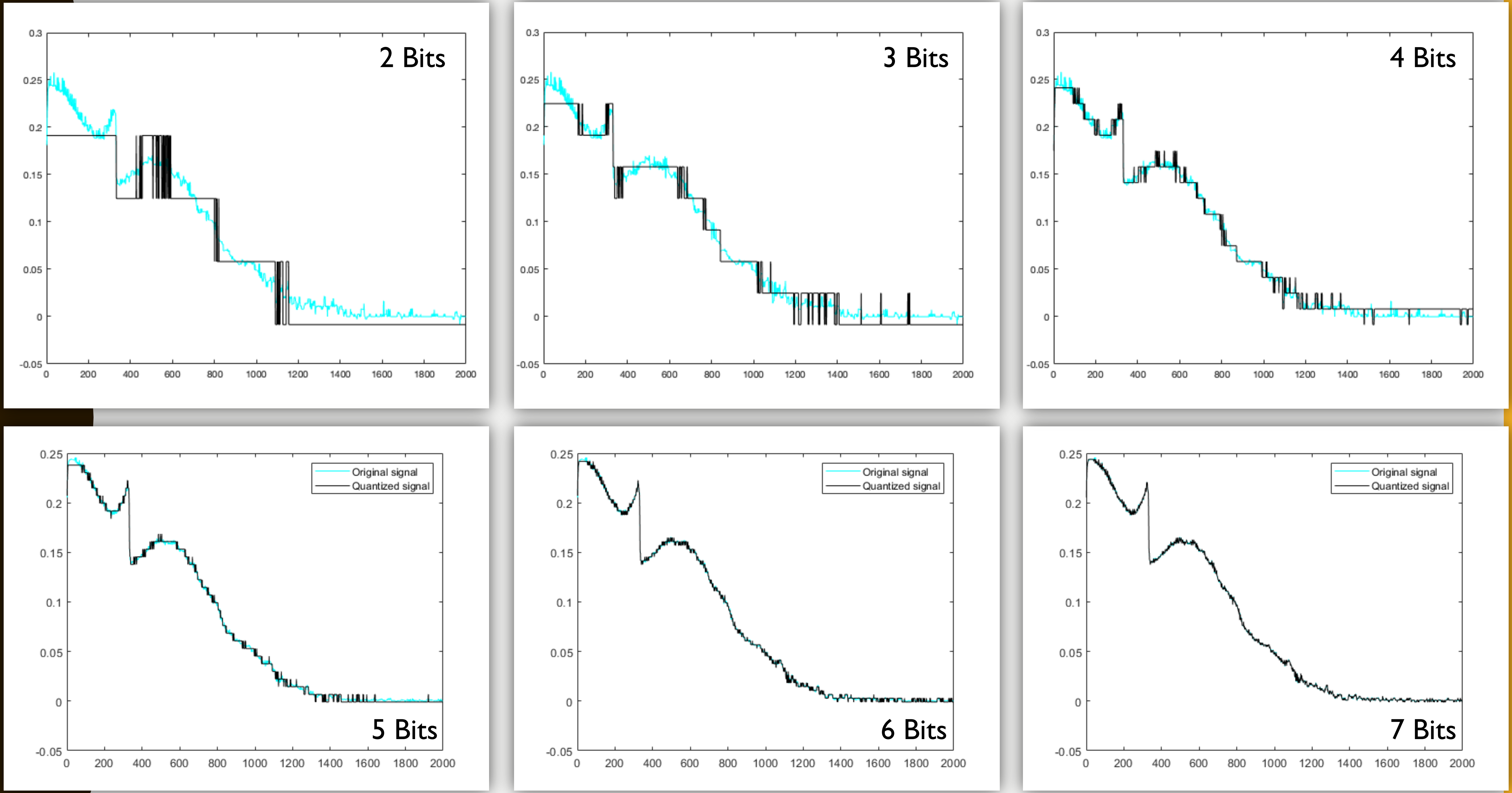
Quantisation is used to convert sensor sequences into **binary**!

This enables us to:

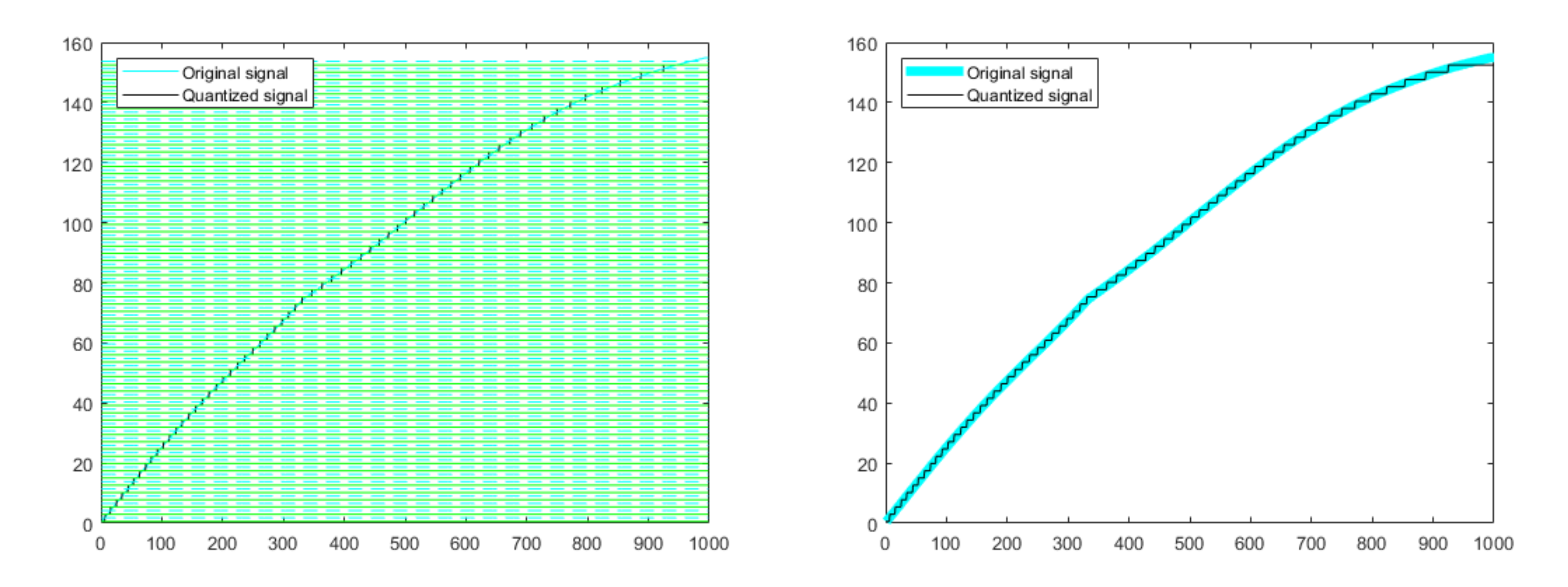
- Perform analytics much more easily.
- Compute the *Hamming Distance*.
- Generate key material/fingerprints.



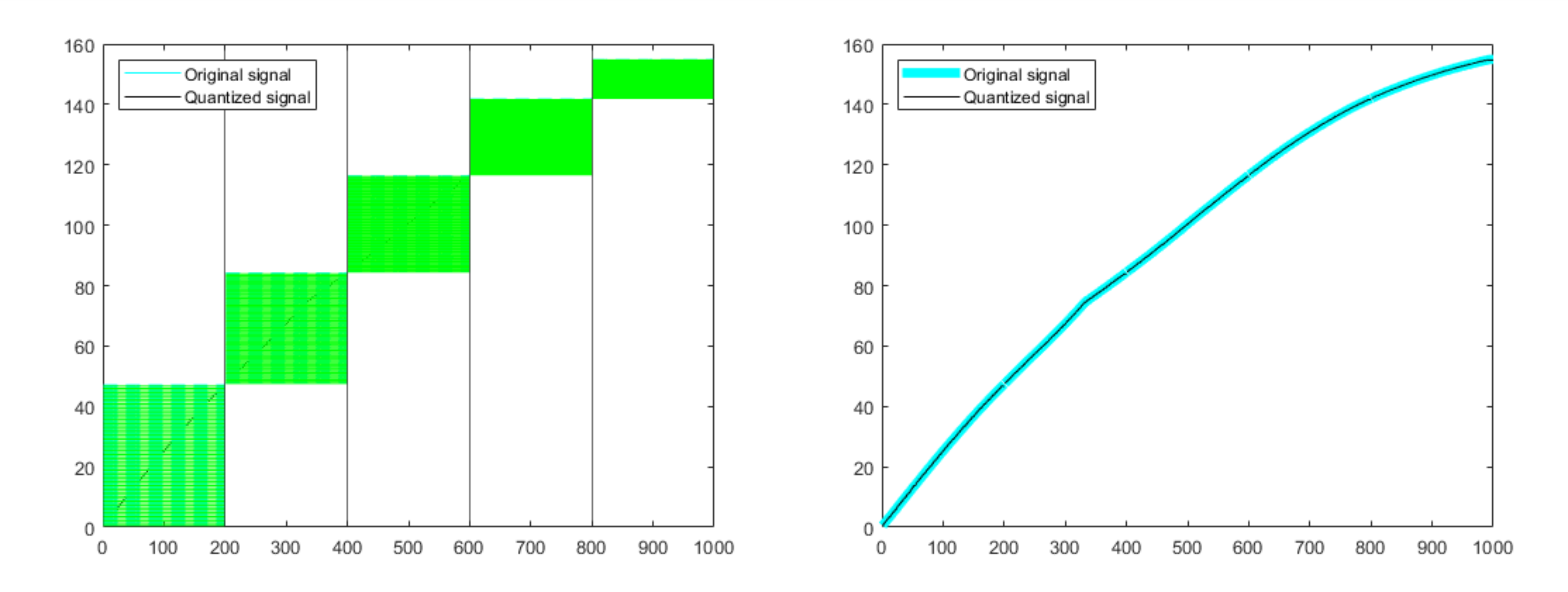
Example of signal quantised at a resolution of 2-bits (1st Derivative)


2 Bits
3 Bits
4 Bits
5 Bits
6 Bits
7 Bits
Original signal
Quantized signal
0.3
0.25
0.2
0.15
0.1
0.05
0
-0.05
0
200
400
600
800
1000
1200
1400
1600
1800
2000

EXAMPLE SIGNAL QUANTISED AT 6-BITS

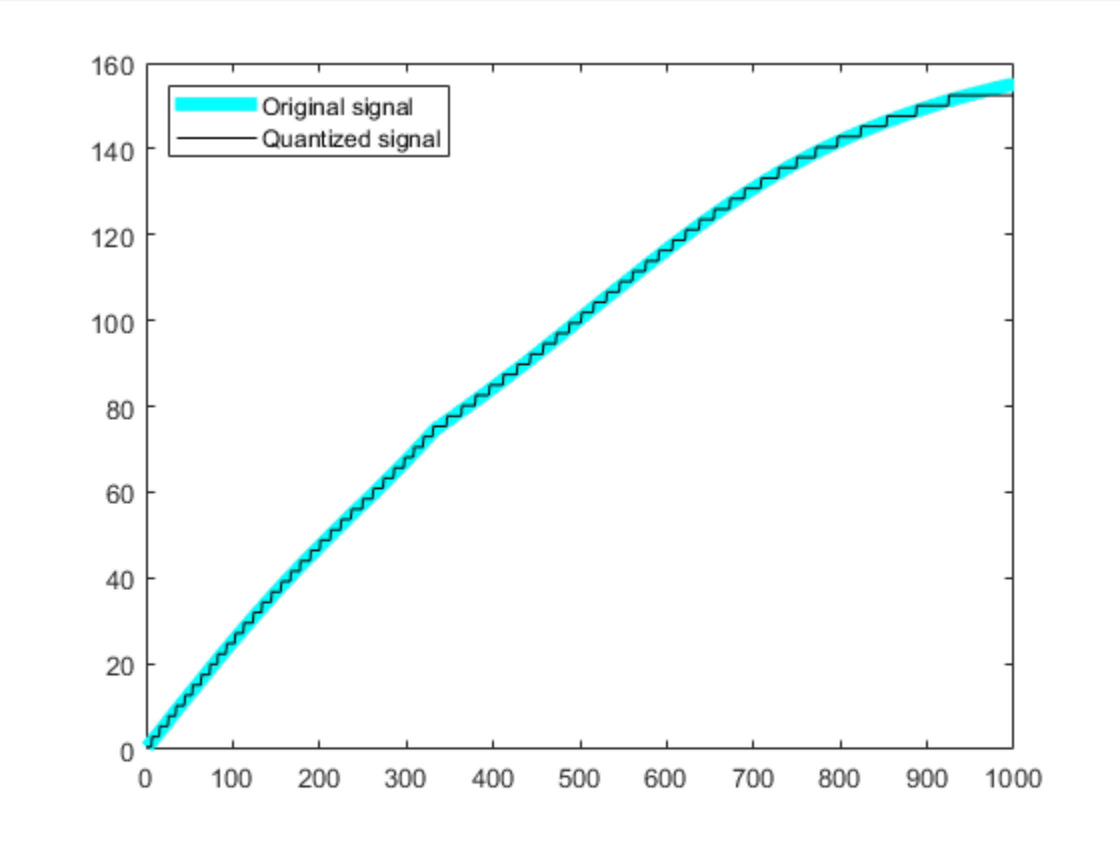


SPLITTING SIGNAL INTO SEGMENTS

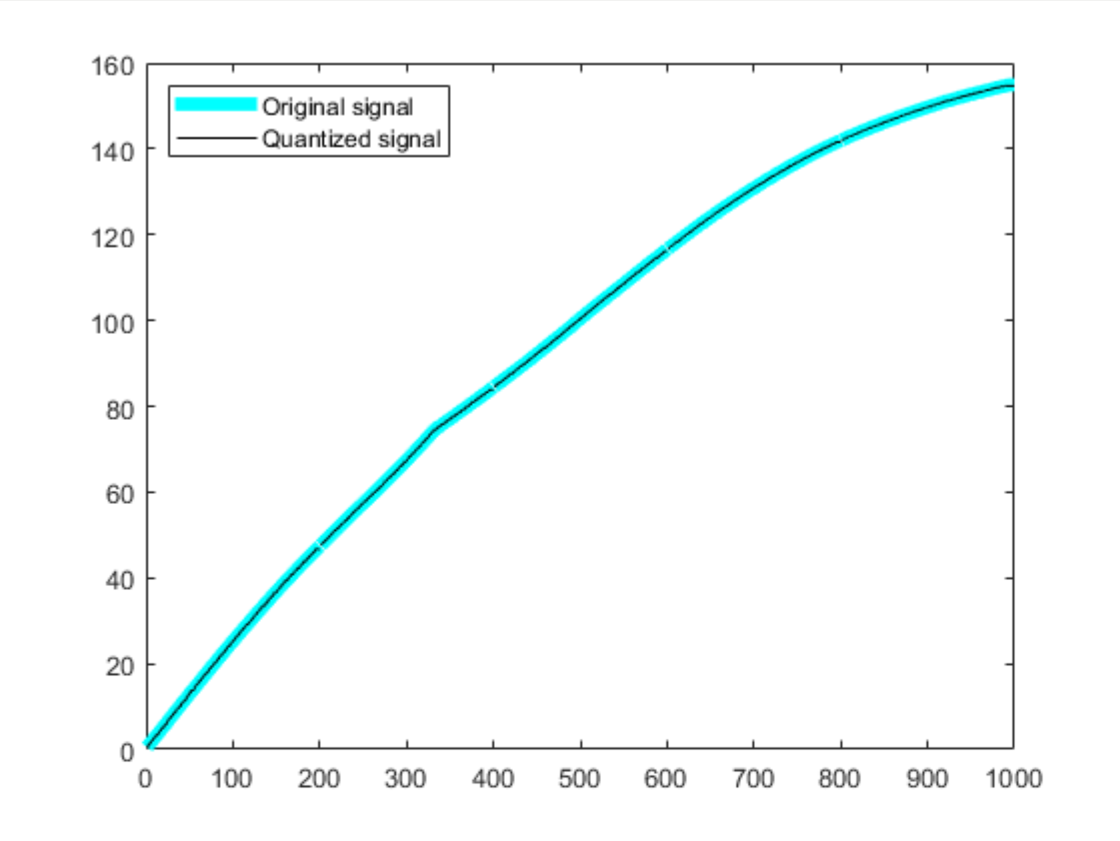


COMPARISON

'STANDARD' QUANTISATION

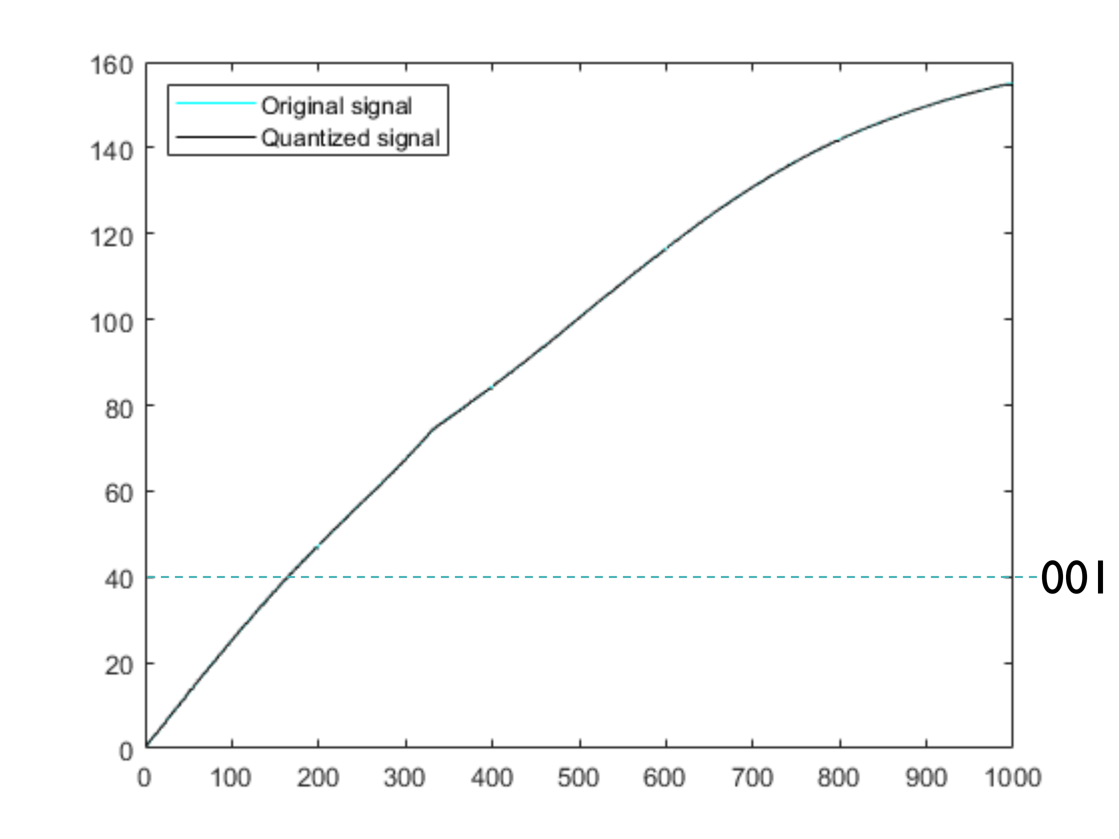


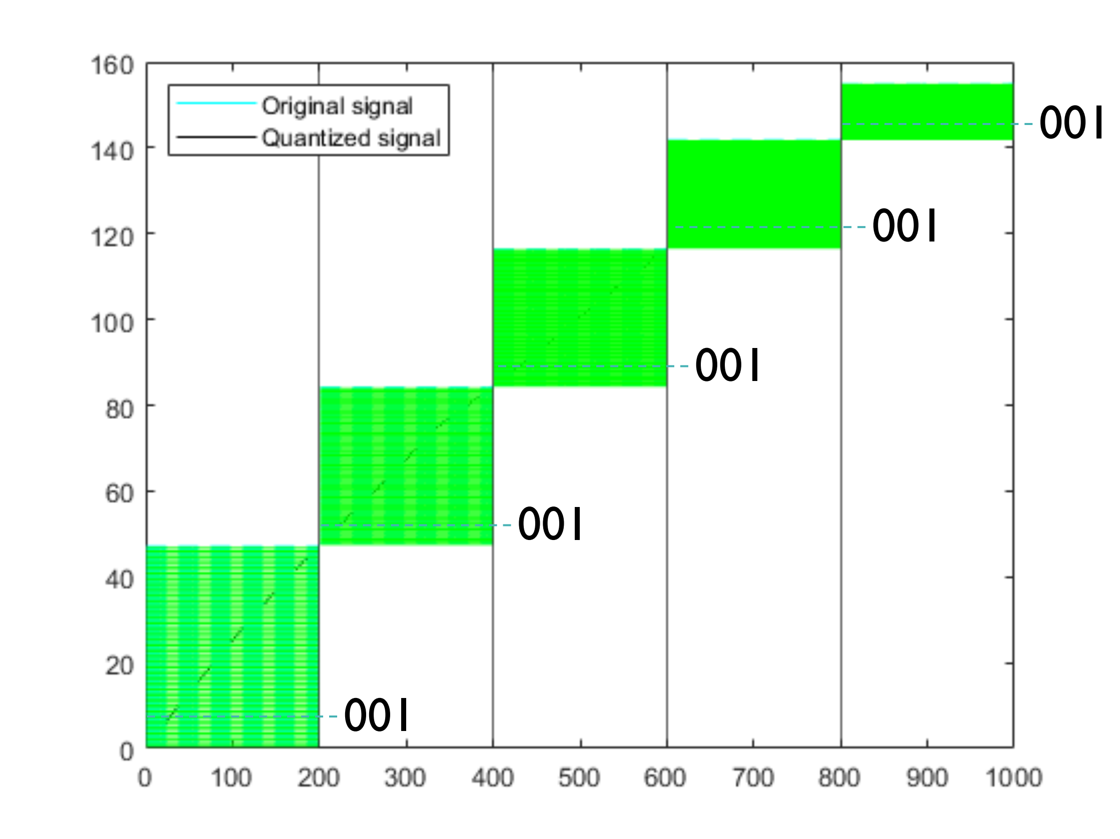
OUR APPROACH



COMPARISON {2}

As our signals are travelling in one direction, certain binary patterns no longer appear after a certain point.





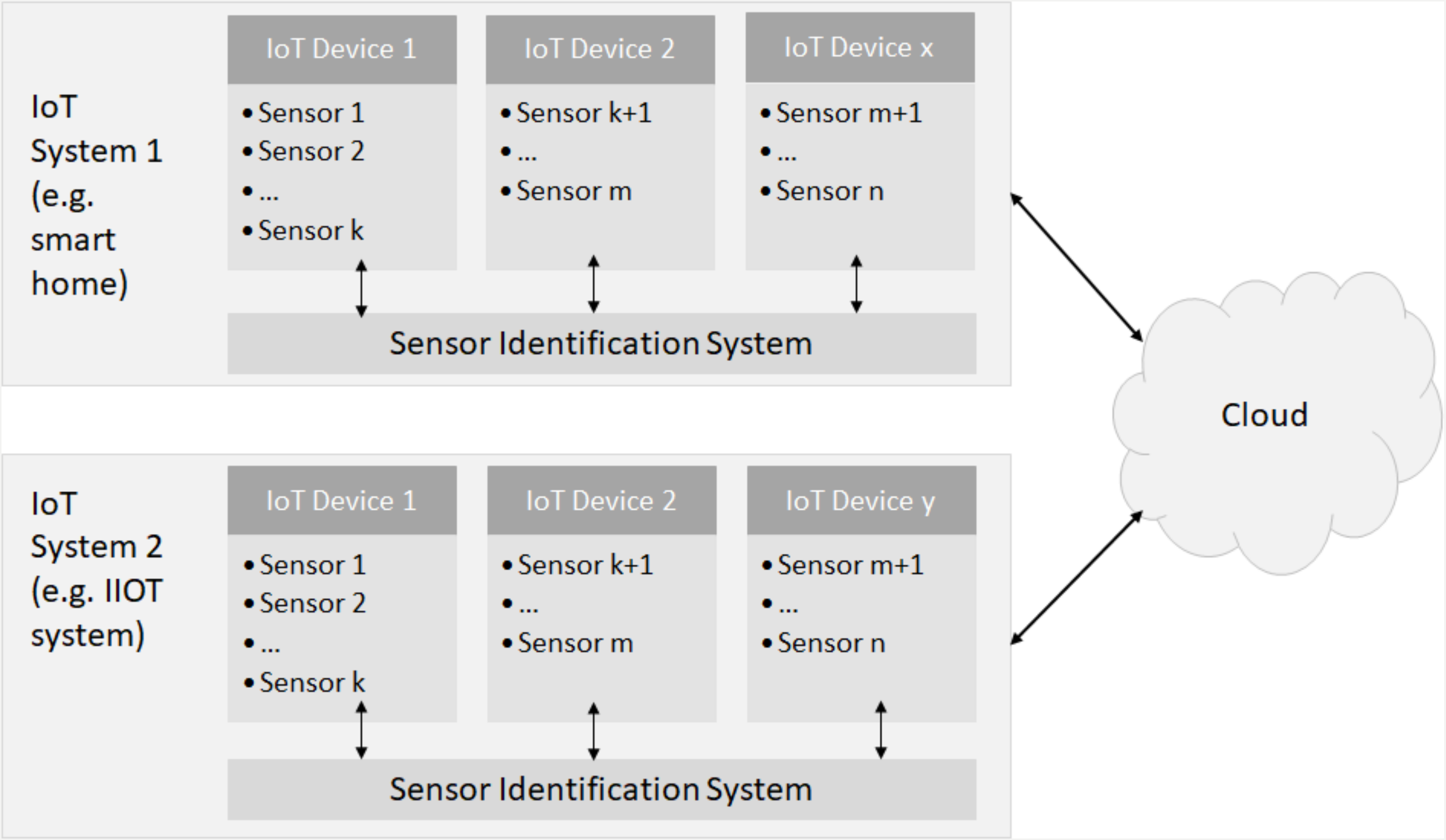
PROOF-OF-CONCEPT IDENTIFICATION SYSTEM

(…AND RESULTS)

CAN OUR FINGERPRINTS ACTUALLY BE USED?

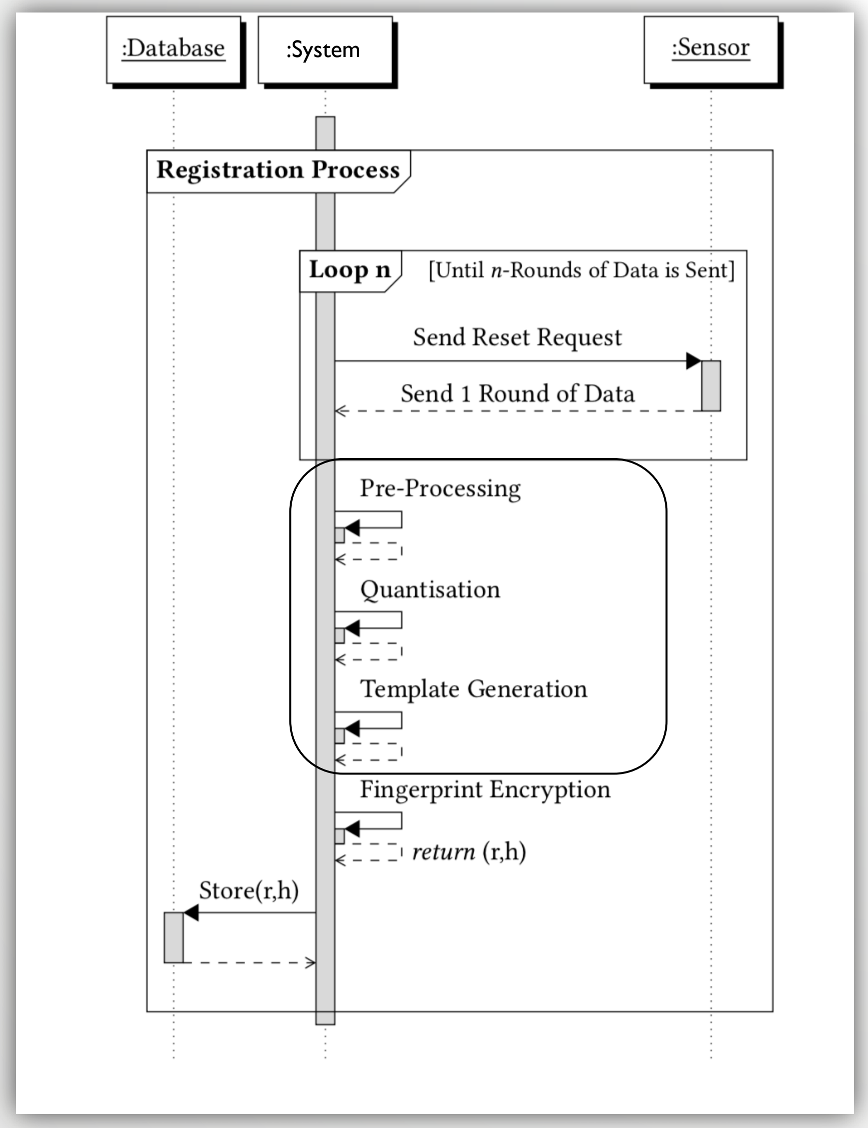
(SPOILER: YES!)

OVERVIEW

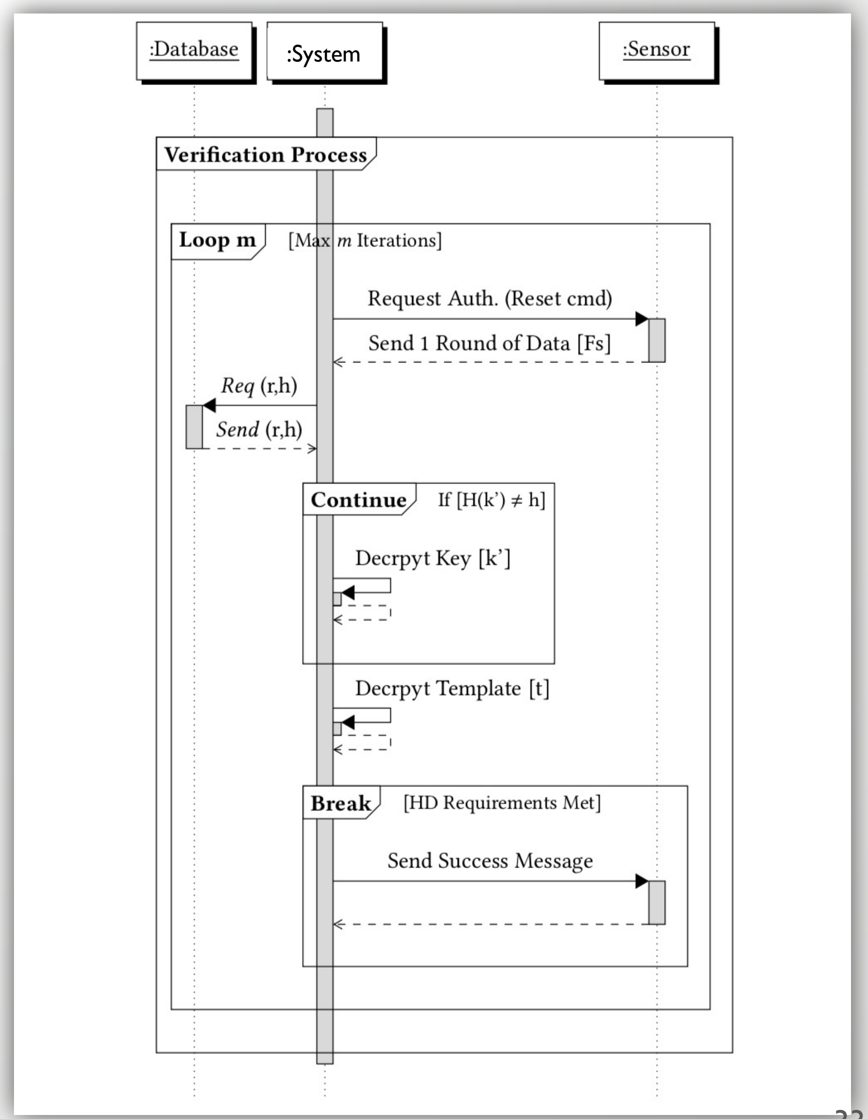


IoT System 1 (e.g. smart home)
IoT Device 1
• Sensor 1
• Sensor 2
• …
• Sensor k
IoT Device 2
• Sensor k+1
• …
• Sensor m
IoT Device x
• Sensor m+1
• …
• Sensor n
Sensor Identification System
IoT System 2 (e.g. IIOT system)
IoT Device 1
• Sensor 1
• Sensor 2
• …
• Sensor k
IoT Device 2
• Sensor k+1
• …
• Sensor m
IoT Device y
• Sensor m+1
• …
• Sensor n
Sensor Identification System
Cloud

REGISTRATION



VERIFICATION



PROOF-OF-CONCEPT PERFORMANCE

Dataset Size: 23 (MPU-6050)

Template Rounds	No. comparisons (Inter, Intra)	Fingerprint Size: 4096 No. False (Accept, Reject)	Fingerprint Size: 2048 No. False (Accept, Reject)
11 (1)	(10,120, 460)	(0, 0)	(0, 0)
11 (2)	(10,120, 460)	(0, 0)	(0, 0)
5 (1)	(13,156, 598)	(0, 0)	(0, 0)
5 (2)	(13,156, 598)	(0, 0)	(0, 0)
5 (3)	(13,156, 598)	(0, 0)	(0, 0)
3 (1)	(14,168, 644)	(0, 1)	(0, 1)*
3 (2)	(14,168, 644)	(0, 0)	(0, 0)
3 (3)	(14,168, 644)	(0, 0)	(0, 0)
1 (1)	(15,180, 690)	(0, 5)	(4, 0)
1 (2)	(15,180, 690)	(0, 4)	(4, 0)
1 (3)	(15,180, 690)	(1, 1)	(0, 2)
1 (4)	(15,180, 690)	(0, 0)	(1, 1)

* Denotes a possible 0% EER upon adjusting threshold

We are able to achieve 0% EER's when using 3 or more rounds of data.

We take this as proof of the effectiveness of an identification system based off of our fingerprints!

ENTROPY

We estimate our solution to contain 38-bits of entropy.

$$N = \frac{\mu(1-\mu)}{\sigma^2}$$

CRITICAL POPULATION SIZE

- Number of sensors which can be individually fingerprinted before a collision is more likely than not.
- Our estimated CPS is **177** and **195** for 2048 and 4096 bit long fingerprints, respectively.
- We have observed an increase in CPS with an increase in dataset size

$$(1 - FMR)^{N(N-1)/2} < 0.5, \quad where\ FMR = FAR \div 100$$

SUMMARY

- We are able to **fingerprint** sensors (gyroscope) through its output alone
- We have proven the feasibility of our fingerprints being used for **identification** purposes
  - 0% **EER** when 3+ rounds of data are used for the template
- We are able to **uniquely** identify up to 195 sensors for a given identification system

FUTURE WORK

- Fingerprinting Different Types of Sensors (Gyroscope, Accelerometer)
- Ageing
- Multiple Sensor Fingerprint
- Real-World Identification System

CURRENT WORK: ICM-20948

All experimentations so far are based on the **mpu-6050**.

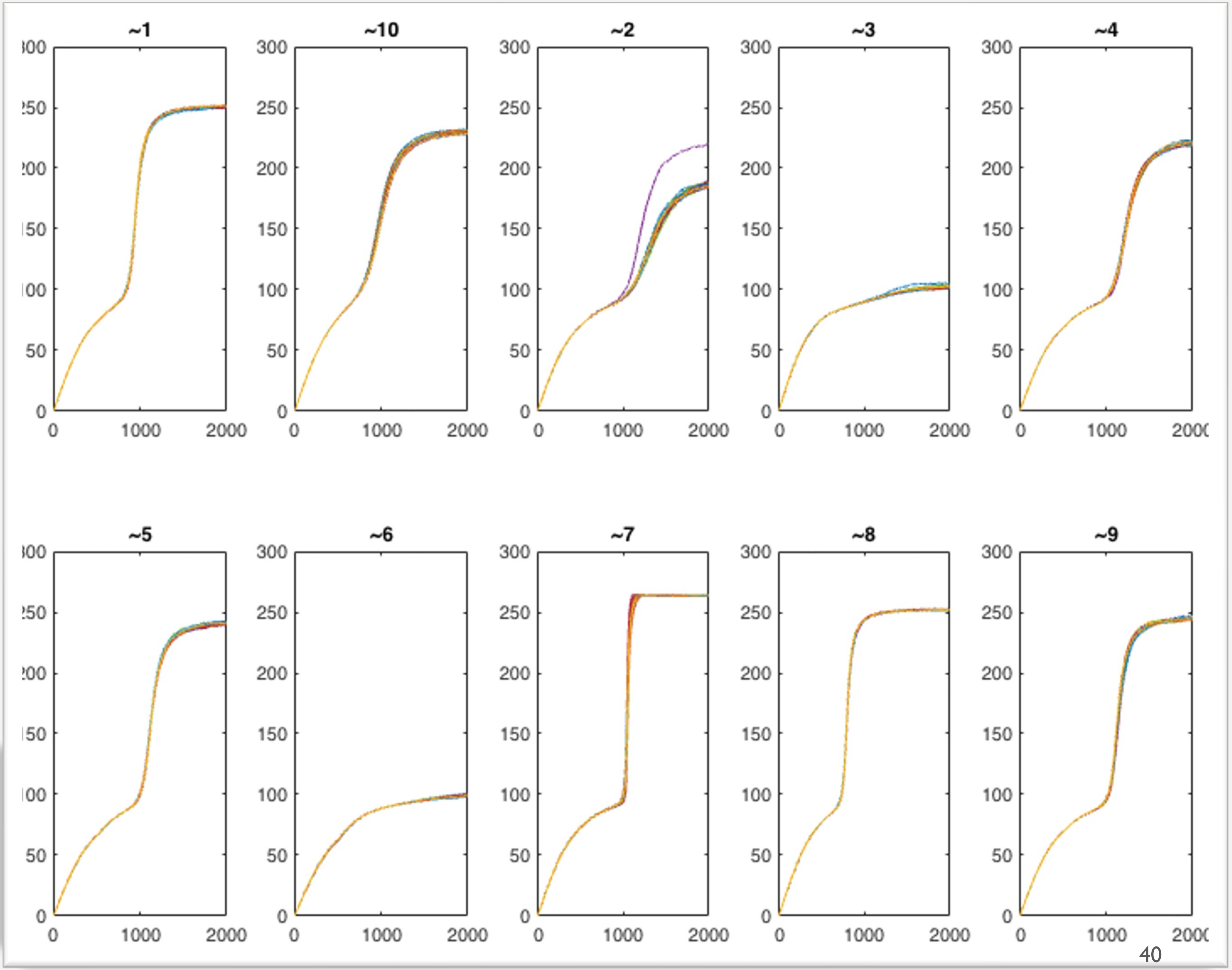
There was a chance of our approach being highly correlated with the output of this IMU.

To address this, we re-ran our experiments on a recently released sensor: *ICM-20948.*

ICM-20948

TIME TO DO IT ALL AGAIN!

ICM-20948 DATASET

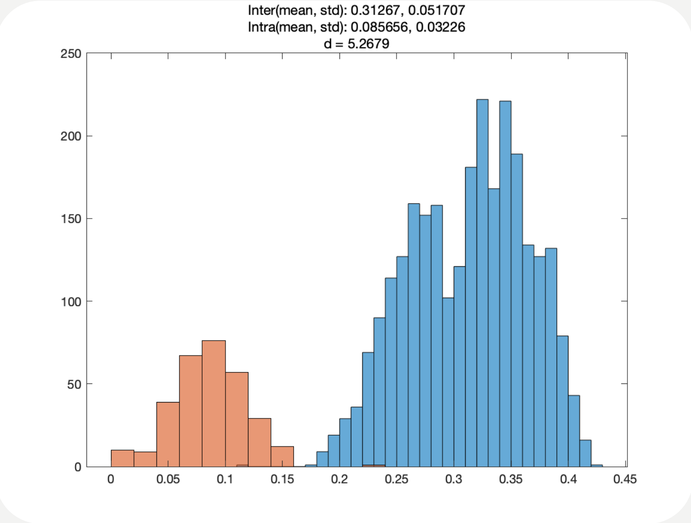


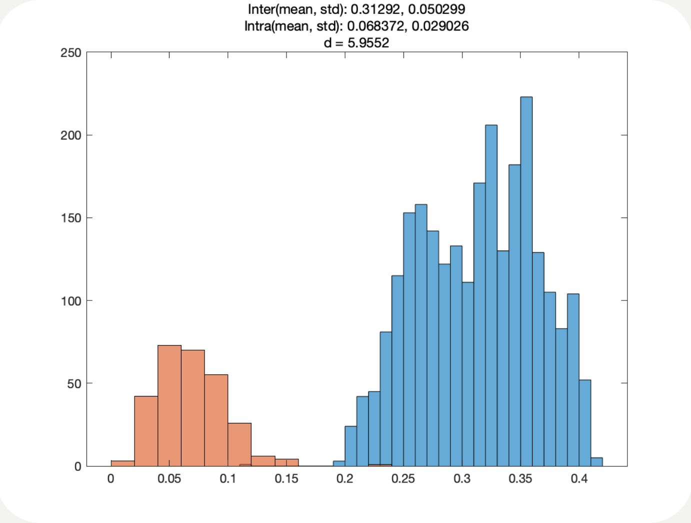
GENERALISATION
(VERY BRIEF SUMMARY)

Experiments were run on a small dataset of 10 sensors

Template Rounds	No.Comparisons(Inter, Intra)	Fingerprint Size	No.False(Accept, Reject)
"1(1)"	2700 300	2048	8 12
"1(2)"	2700 300	2048	2 1
"1(3)"	2700 300	2048	1 1
"1(4)"	2700 300	2048	1 1
"3(1)"	2520 280	2048	1 1
"3(2)"	2520 280	2048	1 1
"3(3)"	2520 280	2048	1 1
"5(1)"	2340 260	2048	1 1
"5(2)"	2340 260	2048	1 1
"5(3)"	2340 260	2048	1 1
"10(1)"	1890 210	2048	1 1
"10(2)"	1890 210	2048	1 1
"10(3)"	1890 210	2048	1 1

Template Rounds	No.Comparisons(Inter, Intra)	Fingerprint Size	No.False(Accept, Reject)
"1(1)"	2700 300	4096	0 34
"1(2)"	2700 300	4096	0 5
"1(3)"	2700 300	4096	0 2
"1(4)"	2700 300	4096	6 9
"3(1)"	2520 280	4096	1 4
"3(2)"	2520 280	4096	1 1
"3(3)"	2520 280	4096	0 2
"5(1)"	2340 260	4096	0 2
"5(2)"	2340 260	4096	0 2
"5(3)"	2340 260	4096	0 1
"10(1)"	1890 210	4096	0 1
"10(2)"	1890 210	4096	0 1
"10(3)"	1890 210	4096	0 1





QUESTIONS, COMMENTS & SUGGESTIONS

THANK YOU!