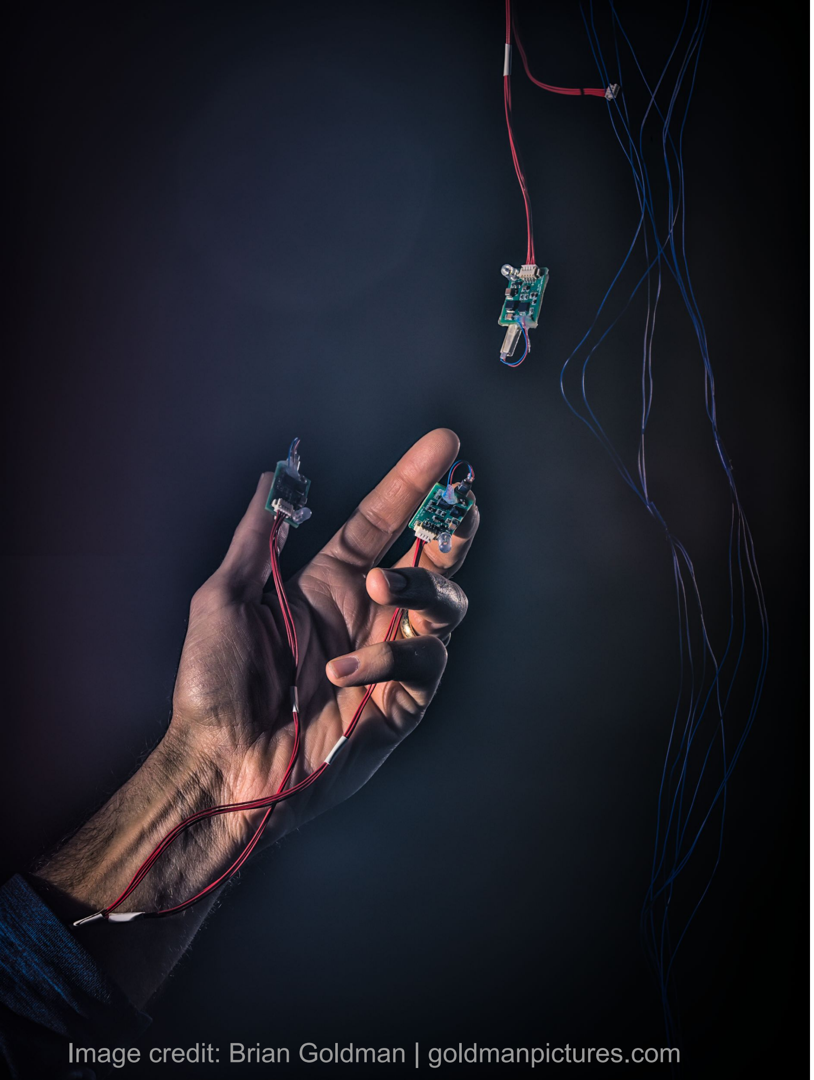


Image credit: Brian Goldman | goldmanpictures.com

neosensory

(non-invasive) Hacking your senses 101

Dr. David M. Eagleman, PhD
Co-Founder, CEO @ Neosensory, Inc.

Dr. Scott Novich, PhD
Co-Founder, CTO @ Neosensory, Inc.

Sensory Augmentation Model

Flow of information

The body

Brain/Central nervous system
Peripheral nervous system
Peripheral sensory receptors

“round hole”

Device

Receptor stimulation

Translation

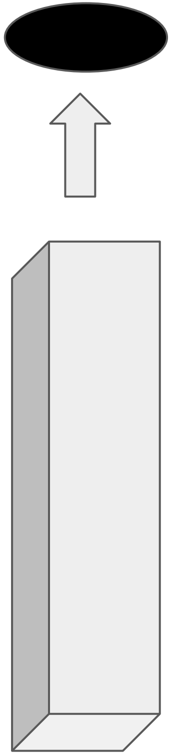
Source capture

“hammer”

The real

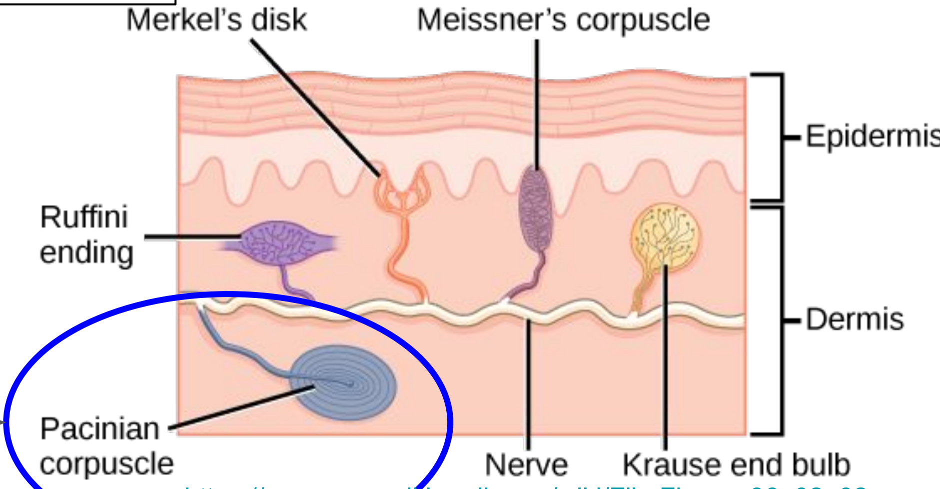
Source information

“square peg”



Example: Neosensory Buzz

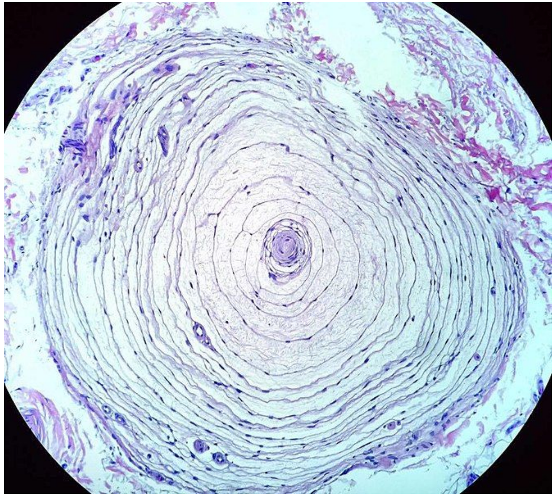
4x Linear Resonant Actuators (LRAs)



https://commons.wikimedia.org/wiki/File:Figure_36_02_02.png

You can read my thesis on more general device design modeling: **https://cutt.ly/sensory-hacking-thesis**

Great video about how Pacinian receptors work: **https://cutt.ly/khan-pacinian-video**



https://commons.wikimedia.org/wiki/File:Pacinian_Corpuscle_(36298105211).jpg

Pacianians fire action potentials proportionally to:

Vibration on 50-300 Hz
Amplitudes of 0-0.1 cm
Can detect **change in vibration as little as 5-10 ms**

(See this paper: https://cutt.ly/pacinian-function)

Buzz LRAs vibrate at a fixed 175 Hz frequency.

Buzz encodes information using **LRA amplitude at 256 discrete levels (8-bits)** around the 0-0.1 cm range.

Buzz LRAs can change intensity at approx 16 ms (also accepted control frame period).

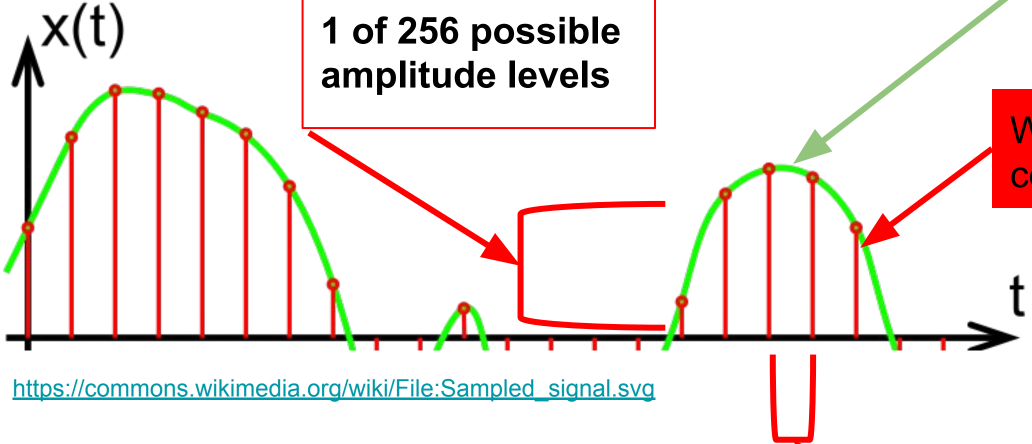
neosensory

Approximating the LRA-Pacinian interface as an analog-to-digital converter

Example signal (i.e. 1-dimensional): like a stock price

1 of 256 possible amplitude levels

What a Pacinian receptor/LRA interface conveys

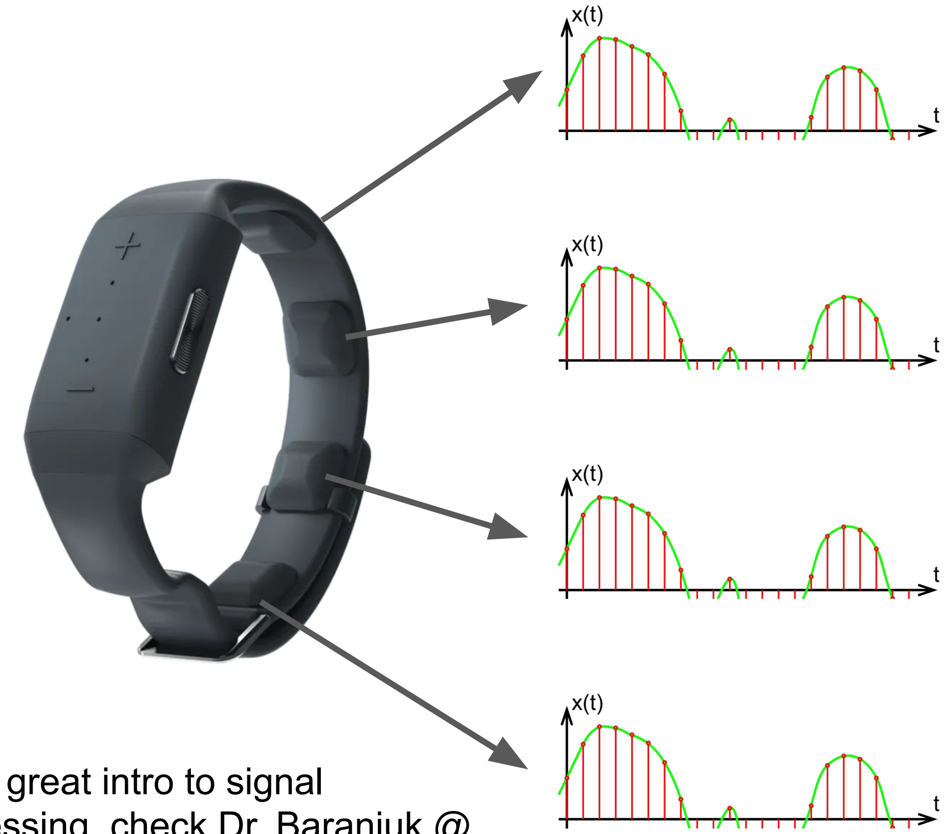


https://commons.wikimedia.org/wiki/File:Sampled_signal.svg

*Can’t provide “negative” vibration

16 ms

You can read my thesis for more thoughts:
https://cutt.ly/sensory-hacking-thesis

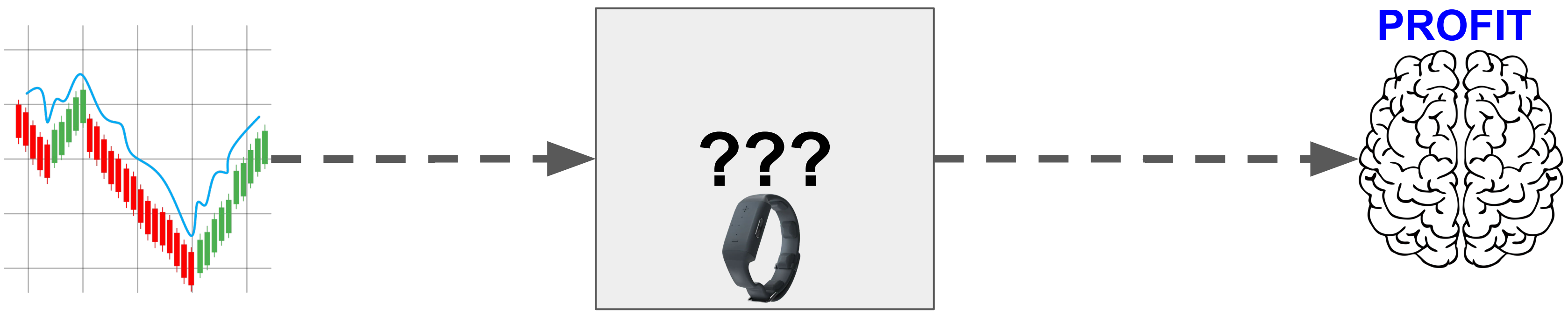


Lay description:
You can encode information across 4 vibration units in 16 ms frames. Each unit can support 256 levels of vibration.

Signal processing description:
Buzz encodes an 8-bit quantized 4D signal with a 62.5 Hz sample rate

For a great intro to signal processing, check Dr. Baraniuk @ RiceU's Signals and Systems (free!): https://cutt.ly/signals-textbook

neosensory



Let’s go with the flow...

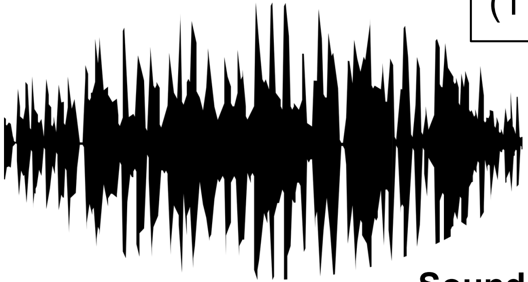
Source Datastream $\xrightarrow{W(t)}$ Feature Extraction/Compression $\xrightarrow{X(t)}$ Rescale I/O dynamic range $\xrightarrow{Y(t)}$ Haptic encoding Z(t)

Source Datastream $\xrightarrow{W(t)}$ Feature Extraction/Compression $\xrightarrow{X(t)}$ Rescale I/O dynamic range $\xrightarrow{Y(t)}$ Haptic encoding $\rightarrow Z(t)$

Types of data!

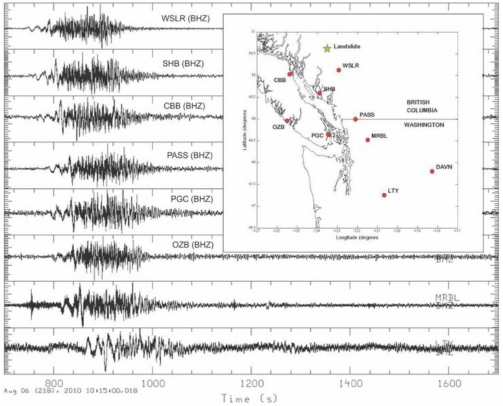
Numerical

1 Dimensional signal
(1 measurement at each timestep)



Sound-wave

N Dimensional signal
(N measurements at each timestep)



Seismic data

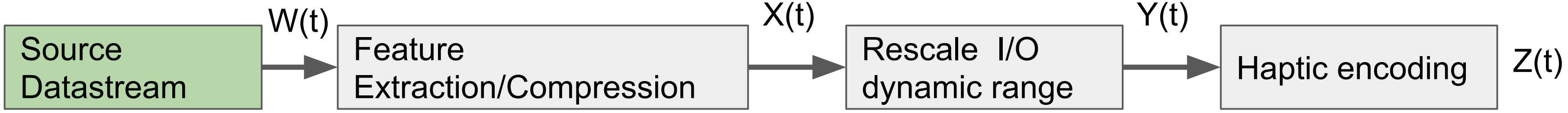
https://commons.wikimedia.org/wiki/File:2010_Mount_Meager_landslide_seismic_signals.png

Categorical

Weather - “nominal”

Emotion - “ordinal”

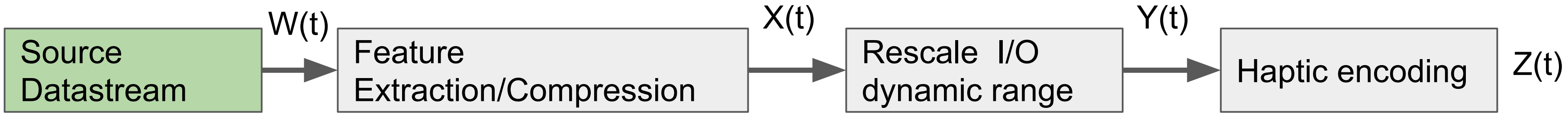
neosensory



Information source tips*

(1) The less latency the better
(2) Information changes at periods < few hundred ms
(3) Your motor output impacts sensory feedback (closed loop system)

* These are just tips. The more strictly you adhere to these, the more powerful the augmentation / potentially less explicit training is needed.



Let’s build a sensory augmentation application!

Before starting: define your goals as precisely as possible (if possible)

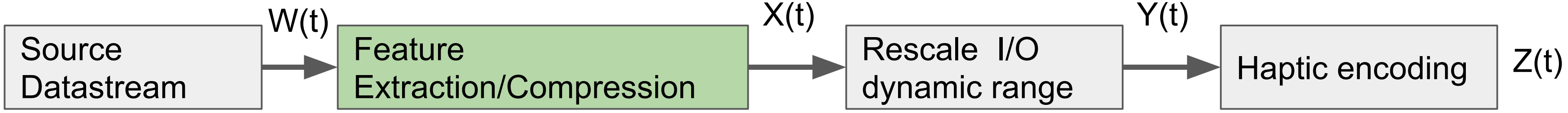
What are the specific task(s) you hope your users can achieve?

“I want to help blind people see”

vs

“I want to help blind people avoid obstacles > **x** cm in size within a radius of **y** meters at heights between **a** and **b** meters”

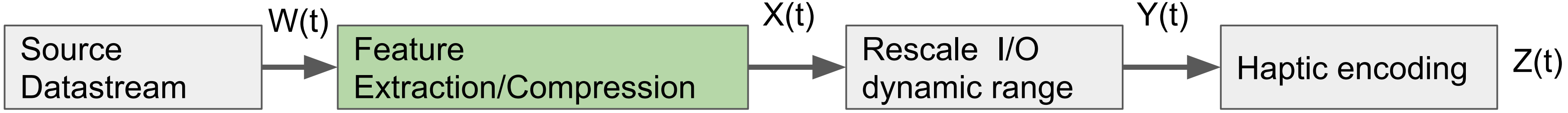
Also: avoid applications that are simply notifications (i.e. something an apple watch could do)



Goals:

(1) Extract only the information we care about.

(2) Compress and re-represent it suitable for Buzz (or your HW)



Example 1: Compass-sense [Easy - source signal fits constraints out of the box]

What makes this nice:

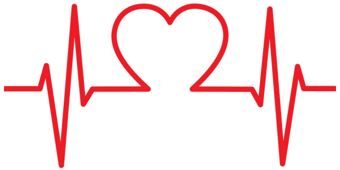
- 1 dimensional numerical signal (your bearing) < 4
- Information you care about can be slower than 16 ms
- You probably don’t need a lot of precision for the compass to be useful (e.g. do you care if you’re heading 312 degrees vs 300 degrees?)
- Bonus: low-latency sensor + sensorimotor I/O

No need to compress the datastream!

neosensory

Source Datastream $\xrightarrow{W(t)}$ Feature Extraction/Compression $\xrightarrow{X(t)}$ Rescale I/O dynamic range $\xrightarrow{Y(t)}$ Haptic encoding $\rightarrow Z(t)$

Example 2: Feeling an electrocardiogram [Trickier - signal too fast for haptic interface]



What makes this nice:

- 1D signal
- Bonus: low latency + potential sensorimotor I/O

What makes this tricky:

- Interesting features of the signal happen at < 16 ms time-scales

Solution: Apply a **signal decomposition** - e.g. a (Discrete) Fourier Transform

or

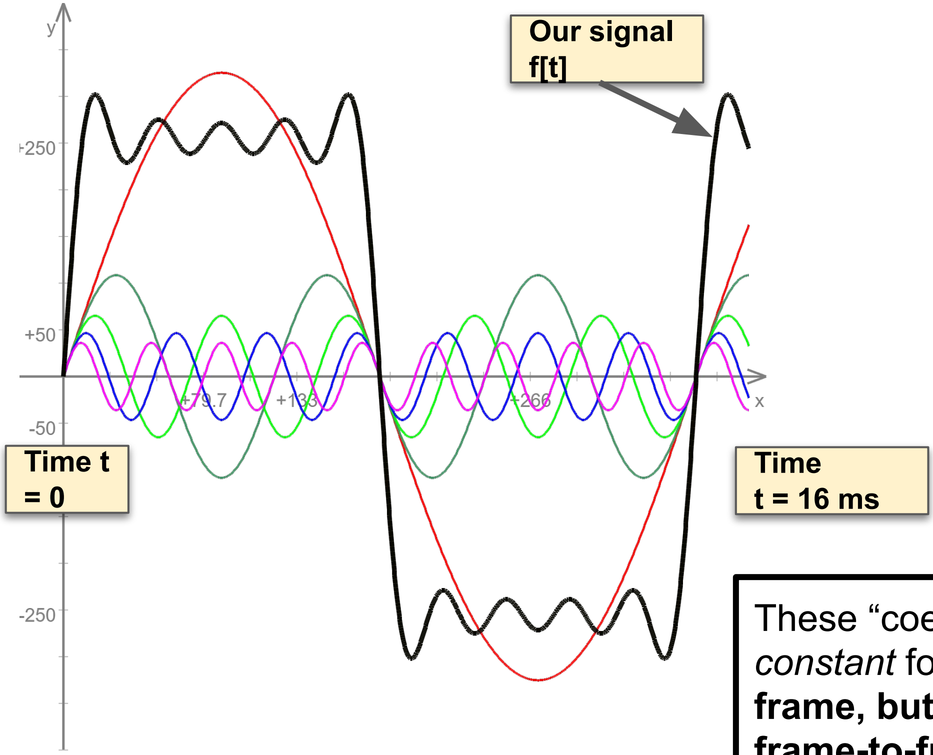
create a filterbank

For a great intro to signal processing, check Dr. Baraniuk @ RiceU's Signals and Systems (free!): https://cutt.ly/signals-textbook

neosensory

A quick detour on signal decompositions

Hack: Trade off speed for dimensionality!



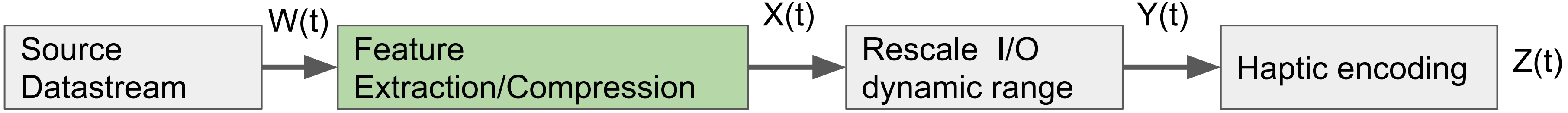
https://commons.wikimedia.org/wiki/File:Fourier_d%27un_carr%C3%A9.svg

16 ms frame

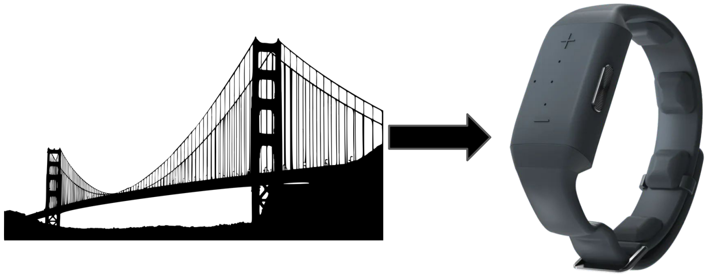
$f[t_{0-15}] =$
$A*sin1[t_{0-15}]$
$+B*sin2[t_{0-15}]$
$+C*sin3[t_{0-15}]$
$+D*sin4[t_{0-35}]$
$+E*sin5[t_{0-35}]$

These "coefficients" are *constant* for the **whole frame, but change frame-to-frame**

These sine functions are exactly the same from from frame-to-frame! Can "ignore"



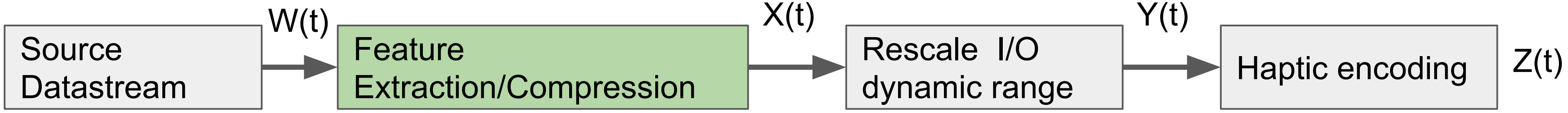
Example 3: Feeling the health of a structure [Tricky - high-dimensional signal]



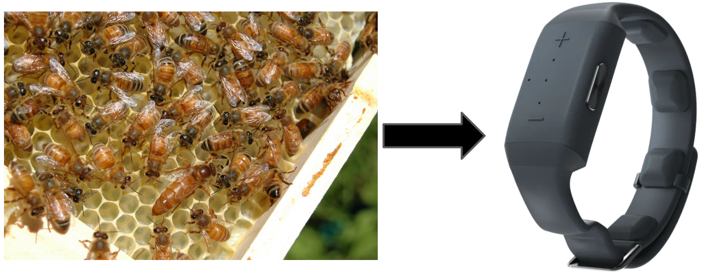
What makes this nice:
- Information you care about likely slower than 16 ms
- Bonus: potentially low-latency signal

What makes this tricky:
- Lots of signals being measured in parallel (> 4)

Apply dimensionality reduction / learn features to get to a 4D output:
PCA
SVD
K-Means Clustering
Autoencoding neural networks
etc.



Example 4: Checking the state of collective bees in your hive (converting to a categorical output)



Let's say, taking a video/audio stream

What makes this nice:
- Information you care about likely slower than 16 ms
- Bonus: potentially low-latency signal

What makes this tricky:
- How can we tell if bees are happy/angry/asleep from complex signals?

Build a machine learning classifier!

EDGE IMPULSE

https://edgeimpulse.com

Edge Impulse is a free platform that makes training and deploying neural networks for edge devices easy!

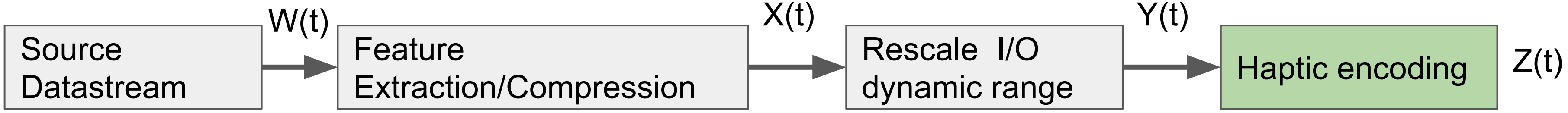
neosensory

Source Datastream $\xrightarrow{W(t)}$ Feature Extraction/Compression $\xrightarrow{X(t)}$ Rescale I/O dynamic range $\xrightarrow{Y(t)}$ Haptic encoding $\rightarrow$ Z(t)

If X(t) is a numerical signal
And
If we intend to value(s) to vibration unit strength

We need to remap the range of values X(t) can take to the range of values vibration unit strength.

Buzz LRAs take values between 0 (off) and 255 (max)



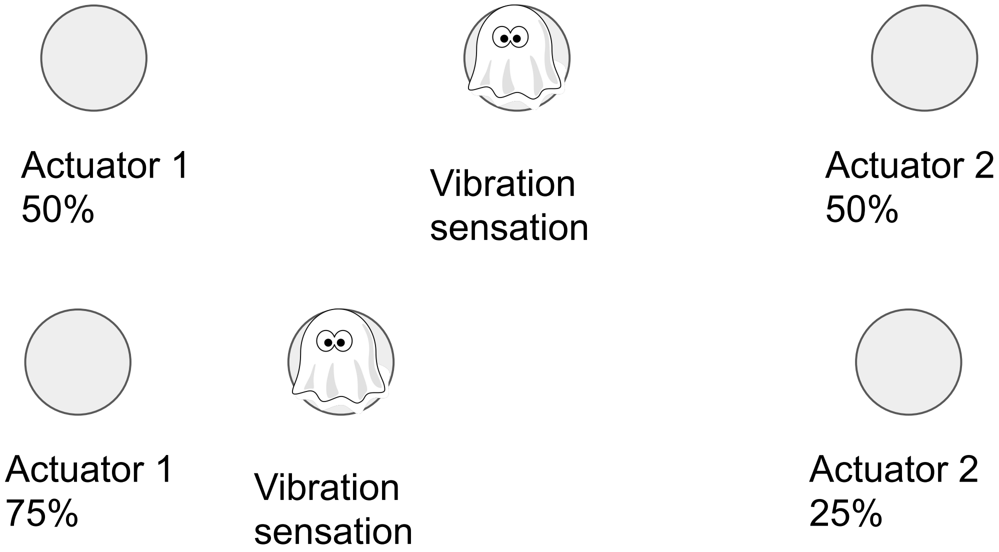
Three dimensions of encoding (for Buzz):

1. Intensity (vibration strength)
2. Space (across 4 LRAs)
3. Time

The more you can leverage all 3, the better.

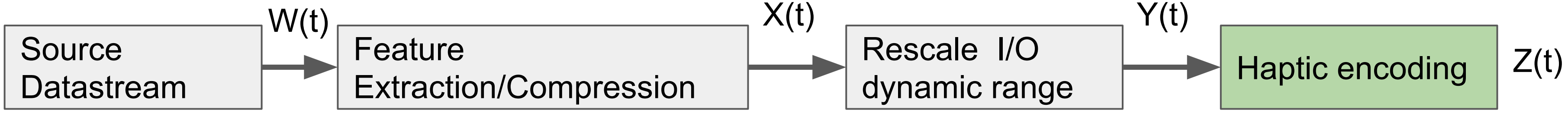
Source Datastream $\xrightarrow{W(t)}$ Feature Extraction/Compression $\xrightarrow{X(t)}$ Rescale I/O dynamic range $\xrightarrow{Y(t)}$ Haptic encoding $\rightarrow Z(t)$

Haptic hack: exploding space with this illusion



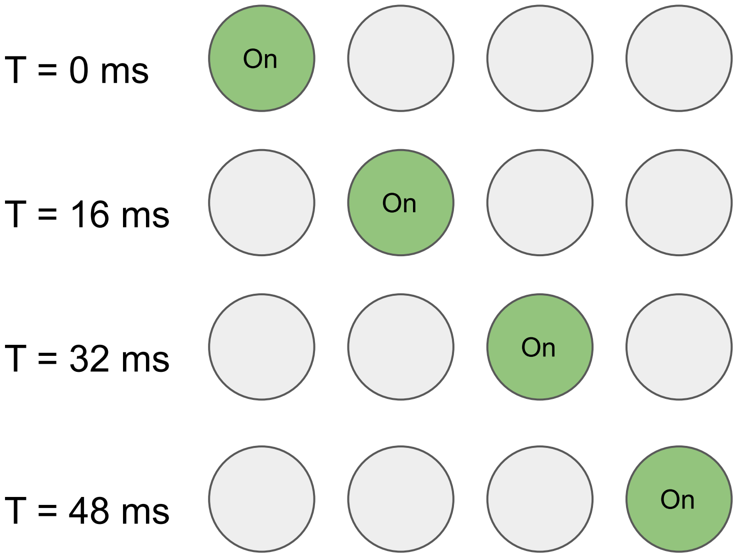
You can also add a “gain” factor to both actuators to make the ghost vibration feel stronger/softer.
Good if want to output a 2D signal with a spatial component.
E.g. direction and how close you’re getting to something.

neosensory

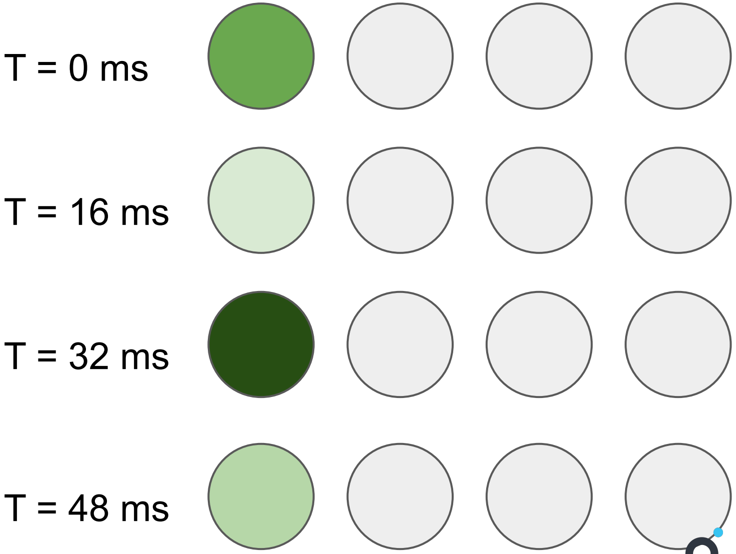


Haptic hack: consider “spatiotemporal” sweeps and textures for **categorical** data

Spatiotemporal sweeps:
Turn adjacent motors on/off in quick succession



Textures:
Modulate vibration intensity of single points and repeat



You’ve prototyped your app. Now what?

RUN EXPERIMENTS!

- Journal your daily usage
- If your app is well-defined enough to help with certain tasks:
  - Build training games to test daily
  - Build test “games” to assess performance at the task

Questions?

Remember to join our contest with edgeimpulse.com at hackster.io!

Applications to win free hardware close this **Friday, November 20th**! (You can always still buy one to participate.)

Webinar resources:

https://neosensory.github.io/sensory_hacking_101_webinar/