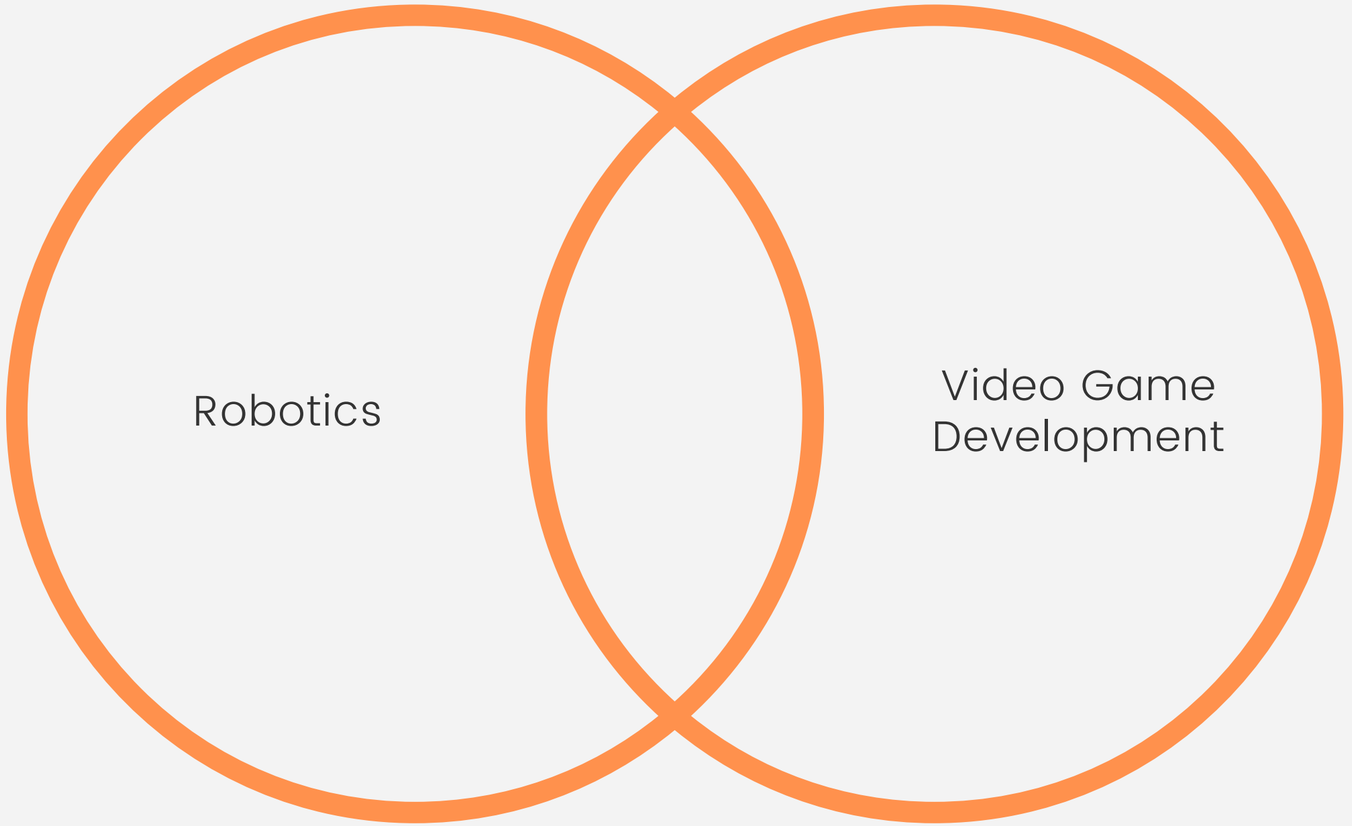
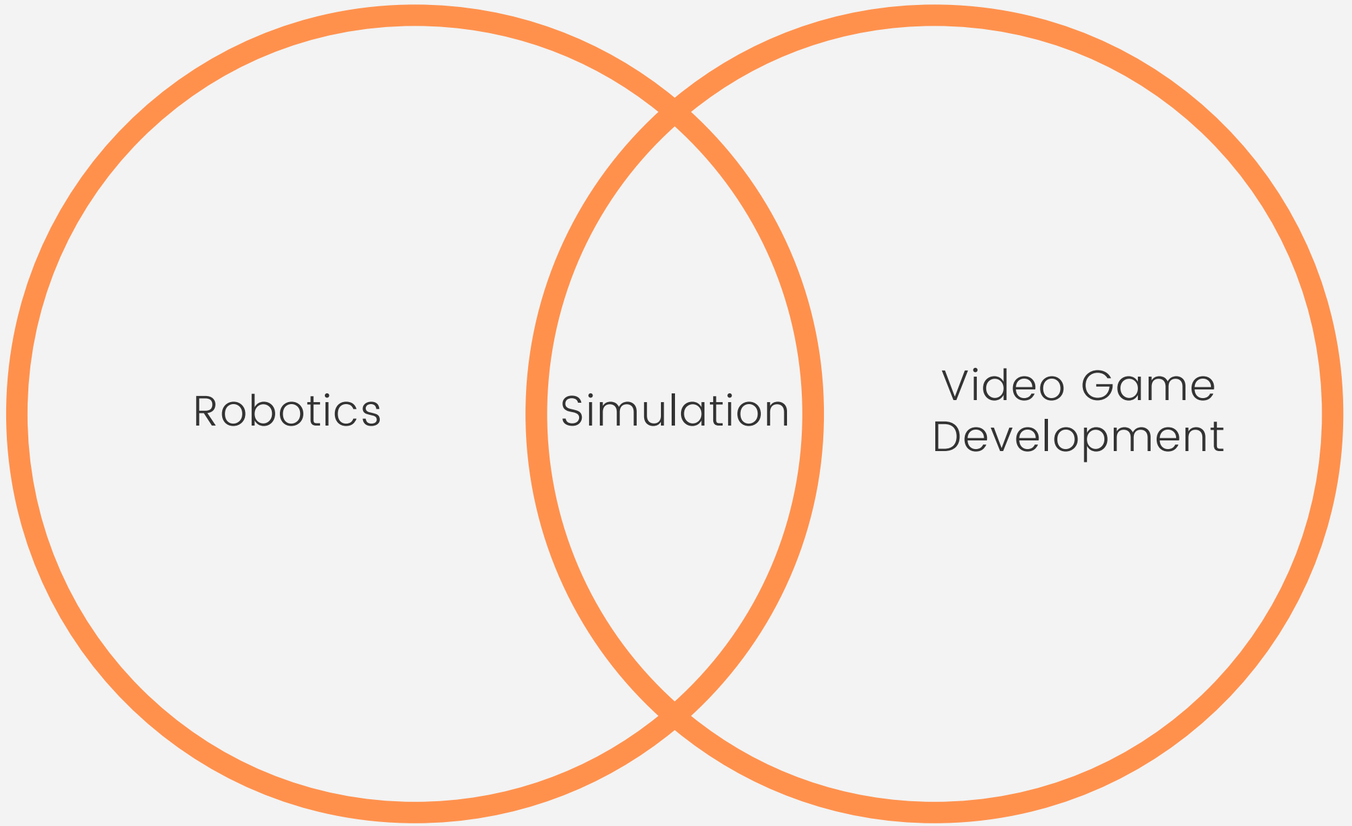
More than Random

Noise Functions

I My Background



I My Background



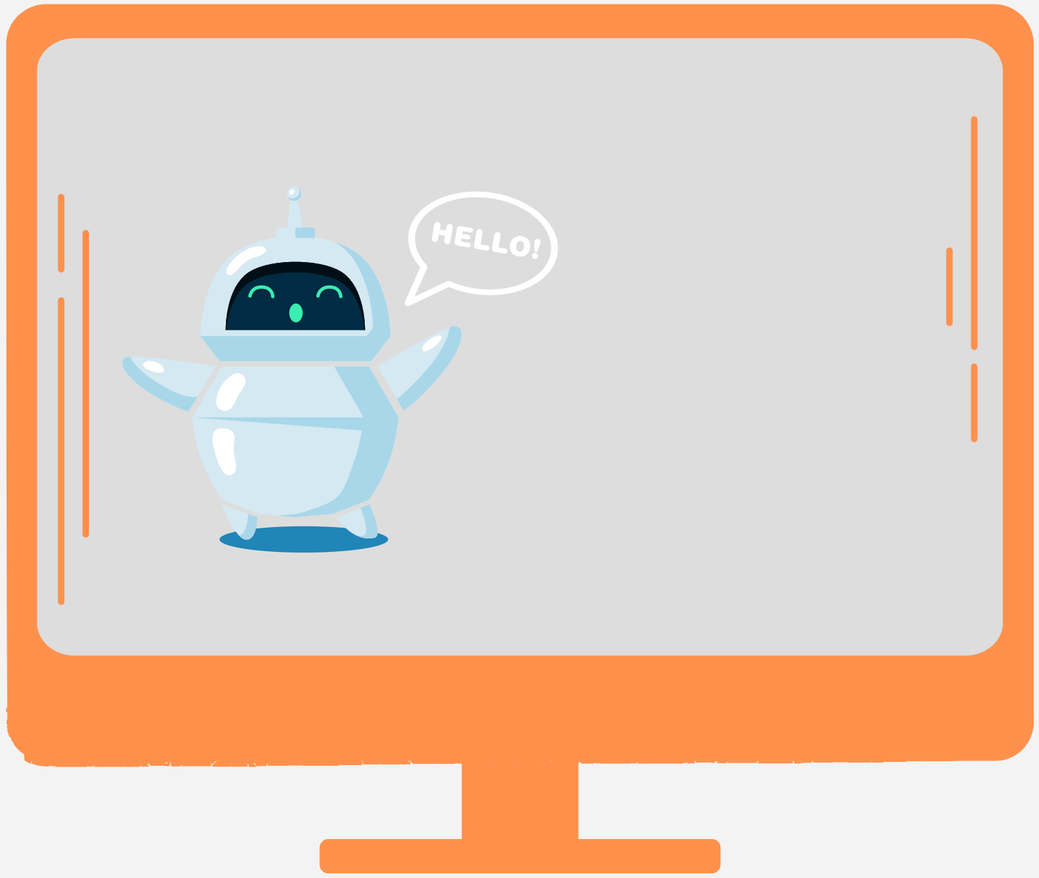
I Goal

Goal:
You are aware of three cool noise functions

Not the Goal:
You understand how these noise functions work

I What is a Noise Function?

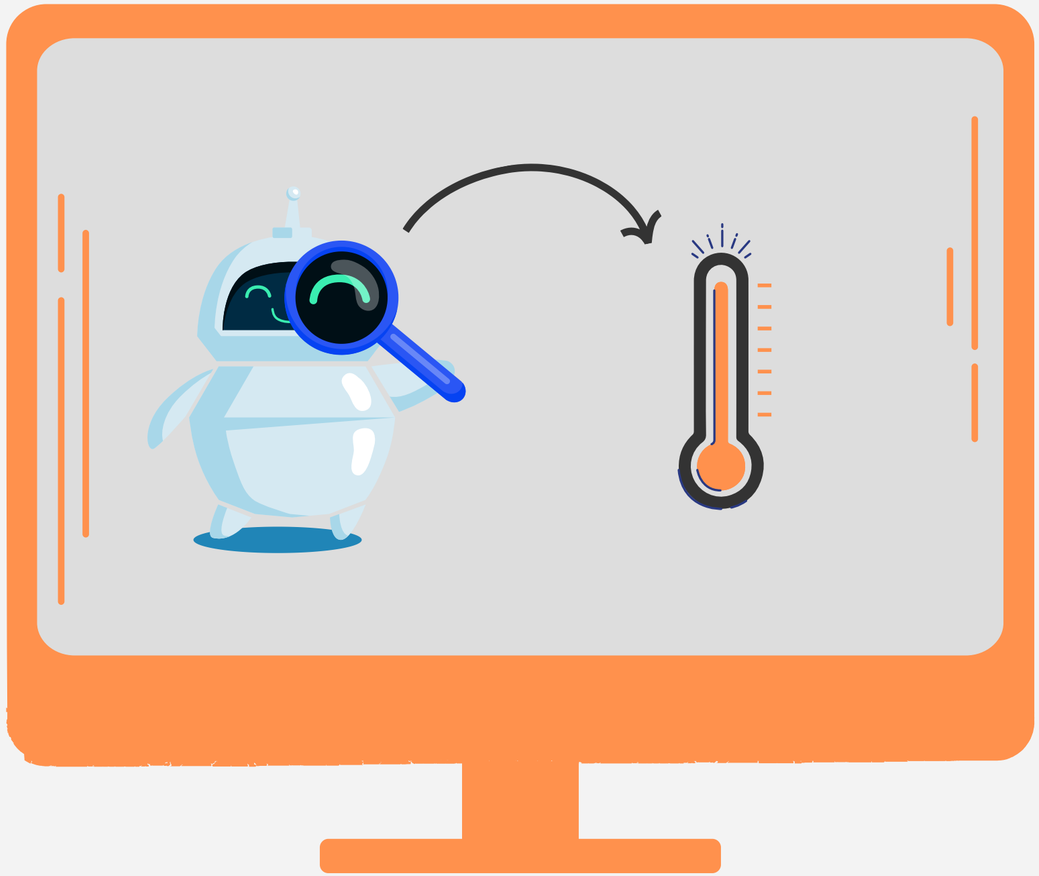
Noise Function:
Means to generate random numbers
(at least in this context)

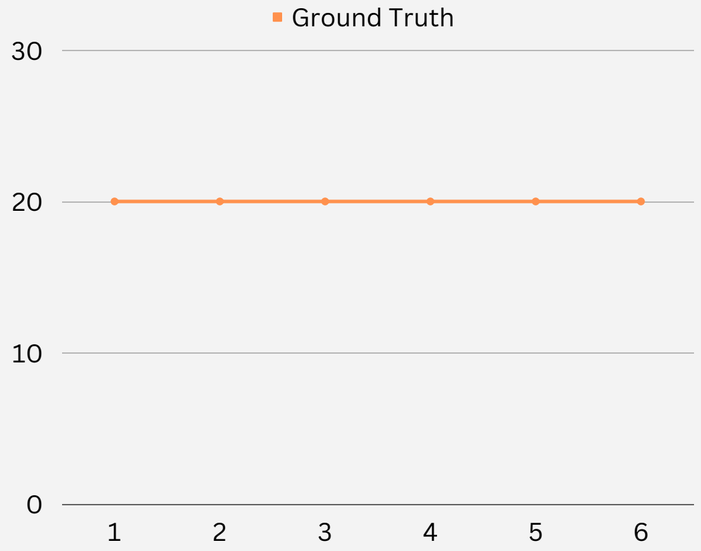


Scenario

Use a simulation to train a robot to solve different tasks

I Random Noise

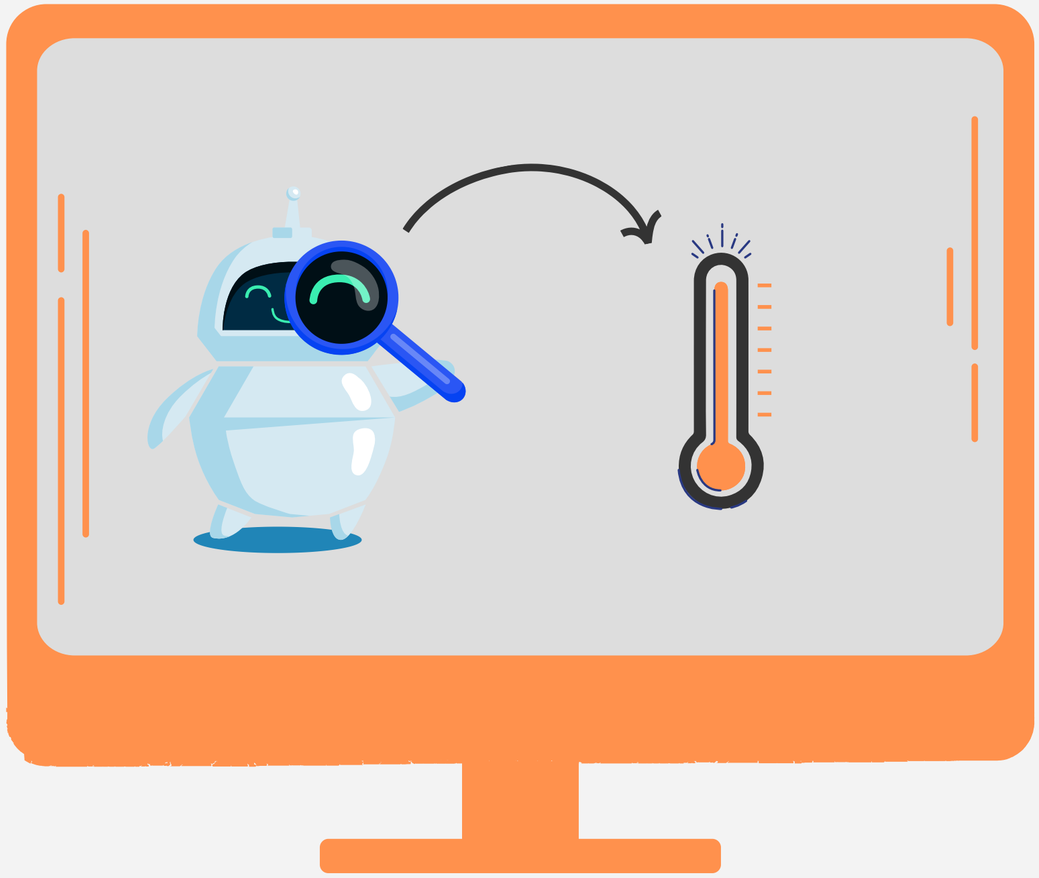


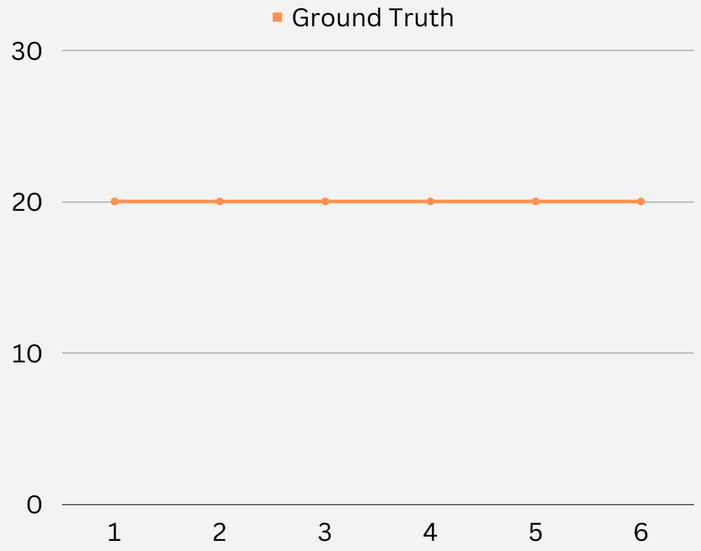


Task

Measure the temperature

I Random Noise

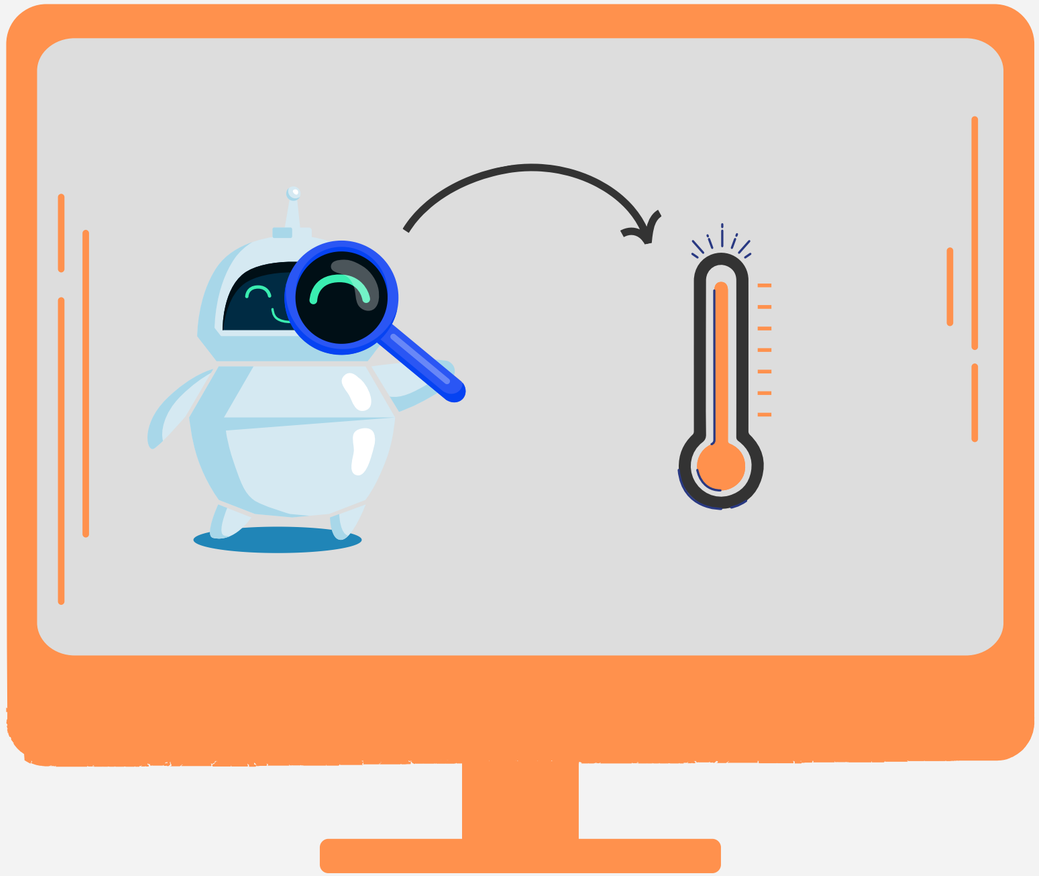


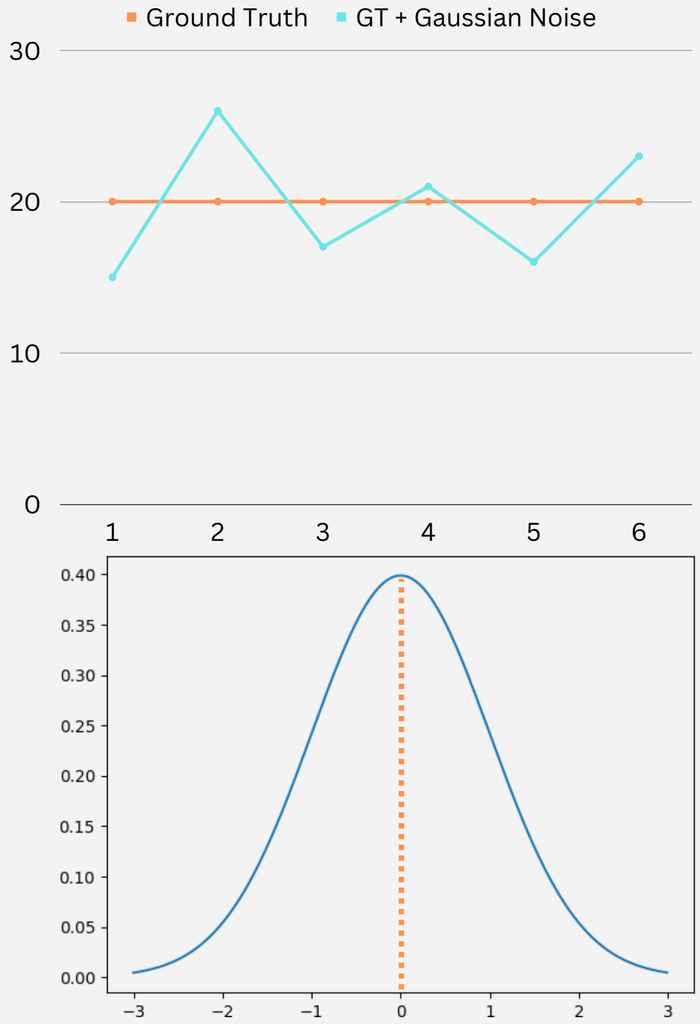


Problem

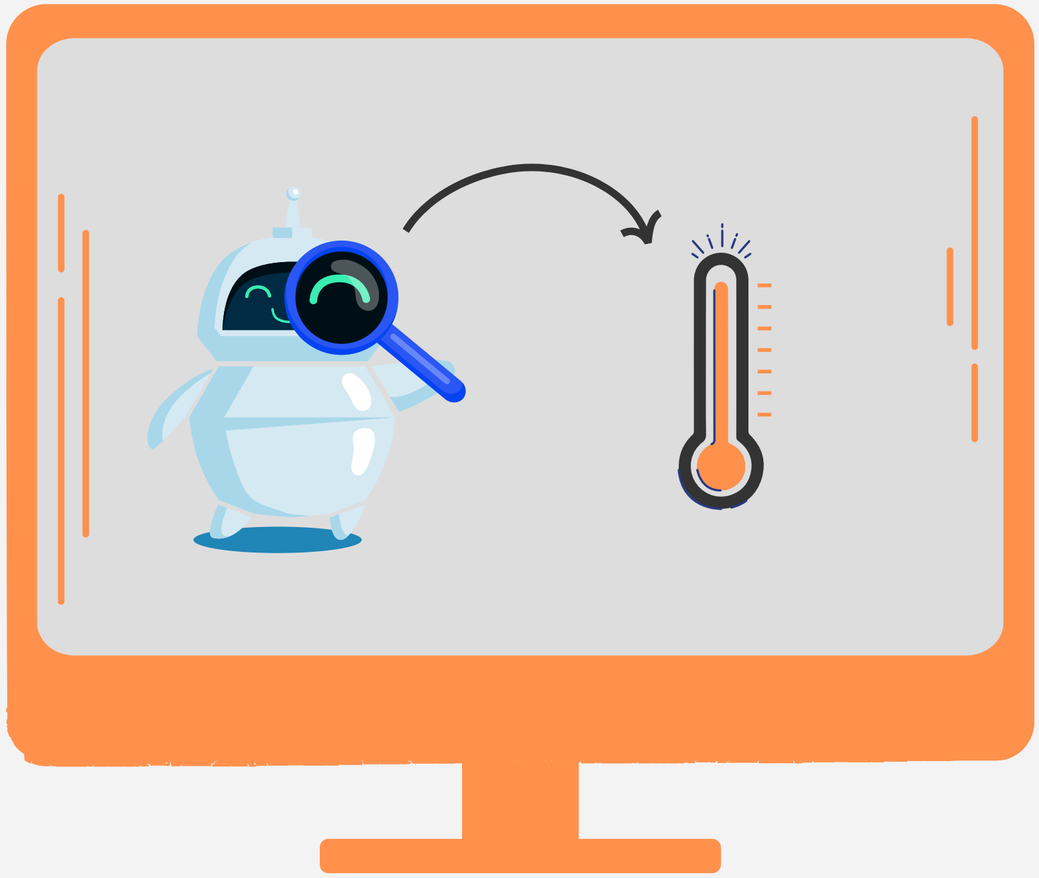
How to model sensor noise?

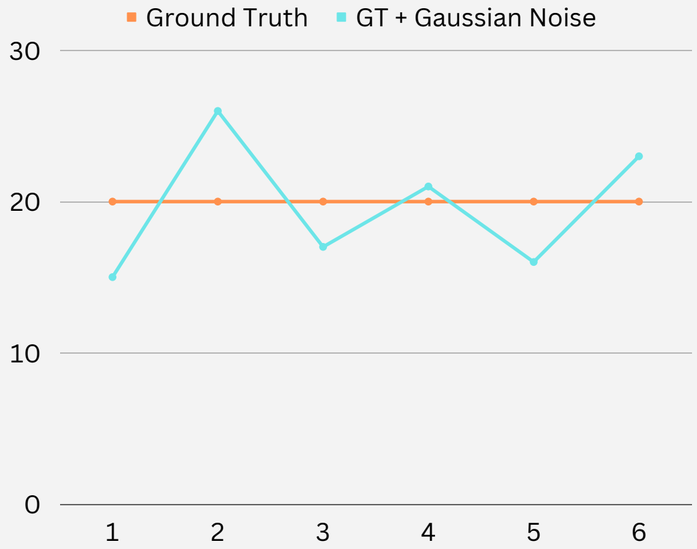
I Random Noise





I Random Noise



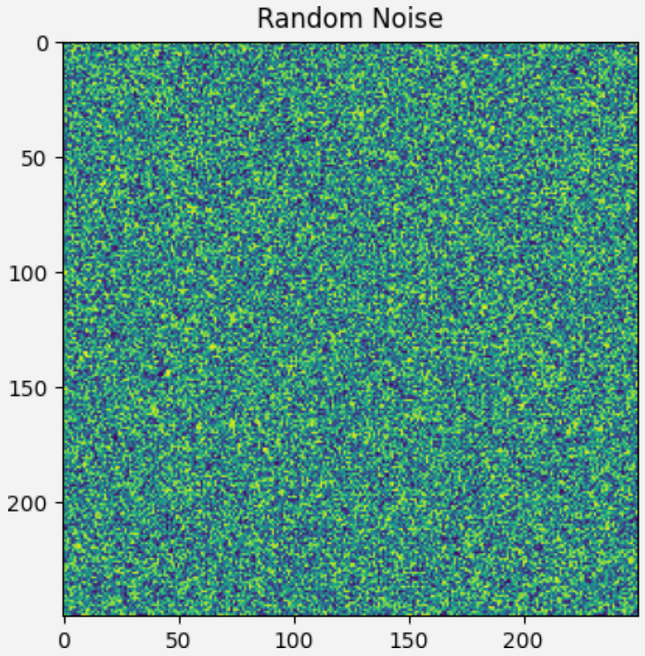


Properties

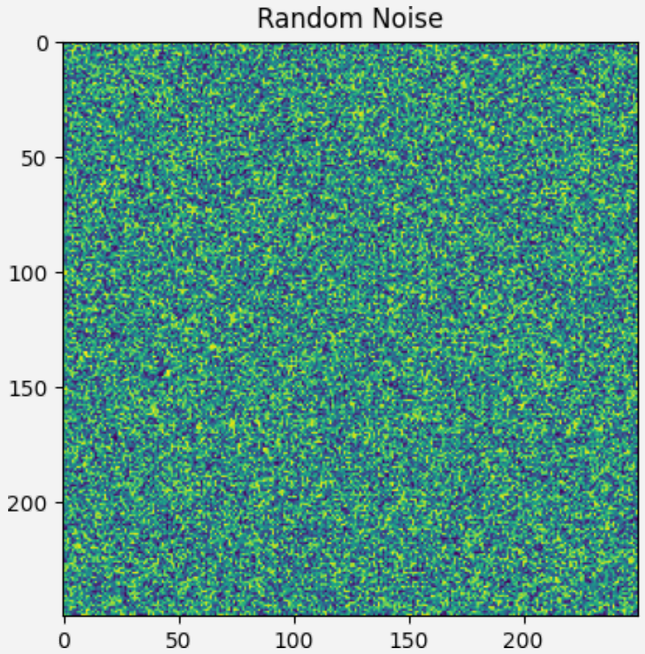
- uncorrelated
- can be averaged out with enough samples

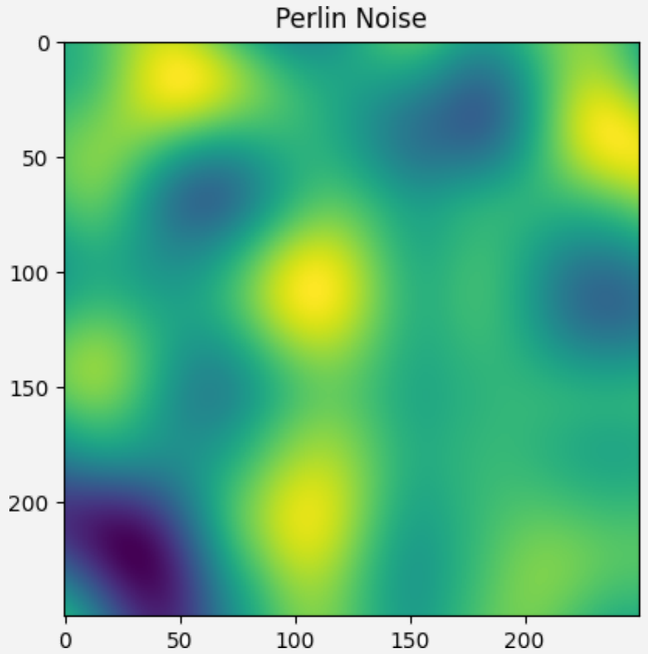
Perlin Noise

II Perlin Noise



II Perlin Noise

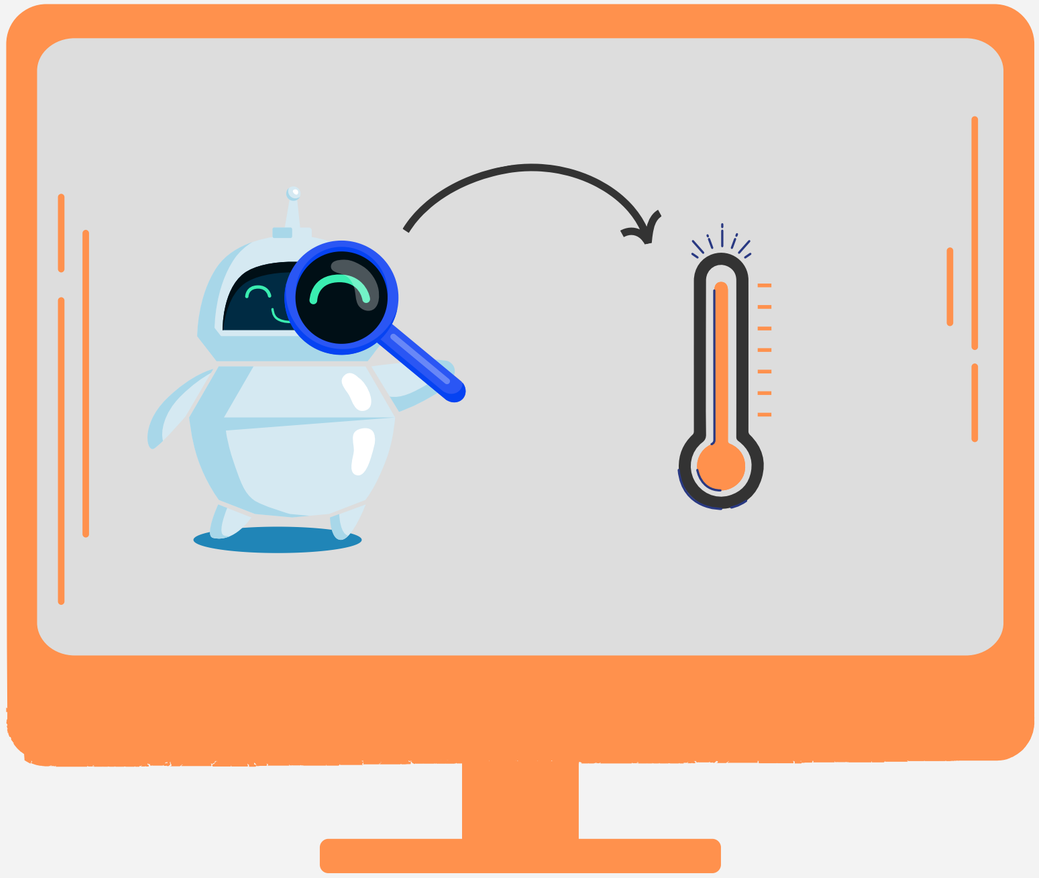


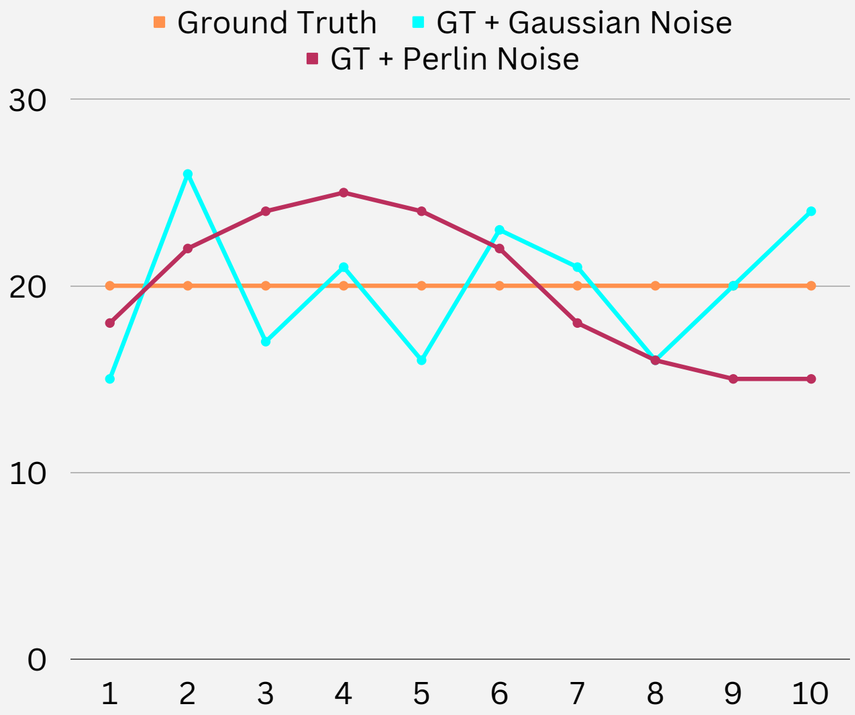


Properties

- Continous
- n-dimensional

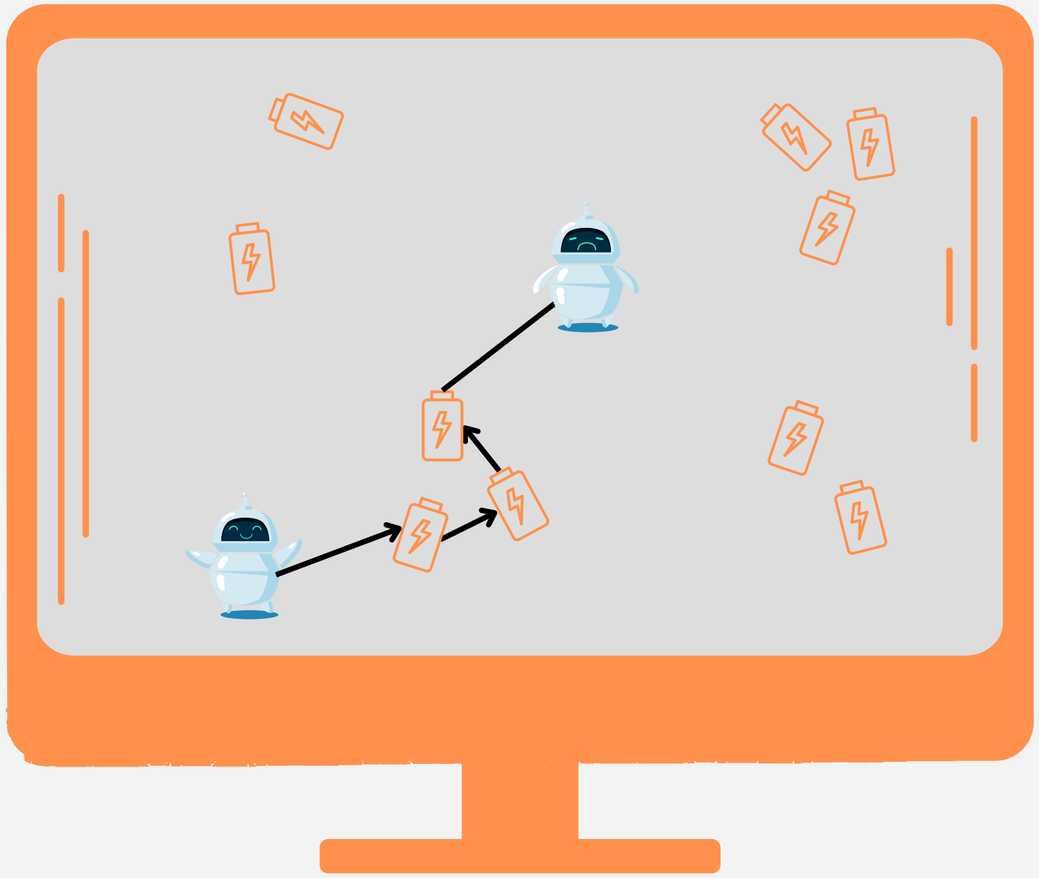
II Perlin Noise





Poission Disk Sampling

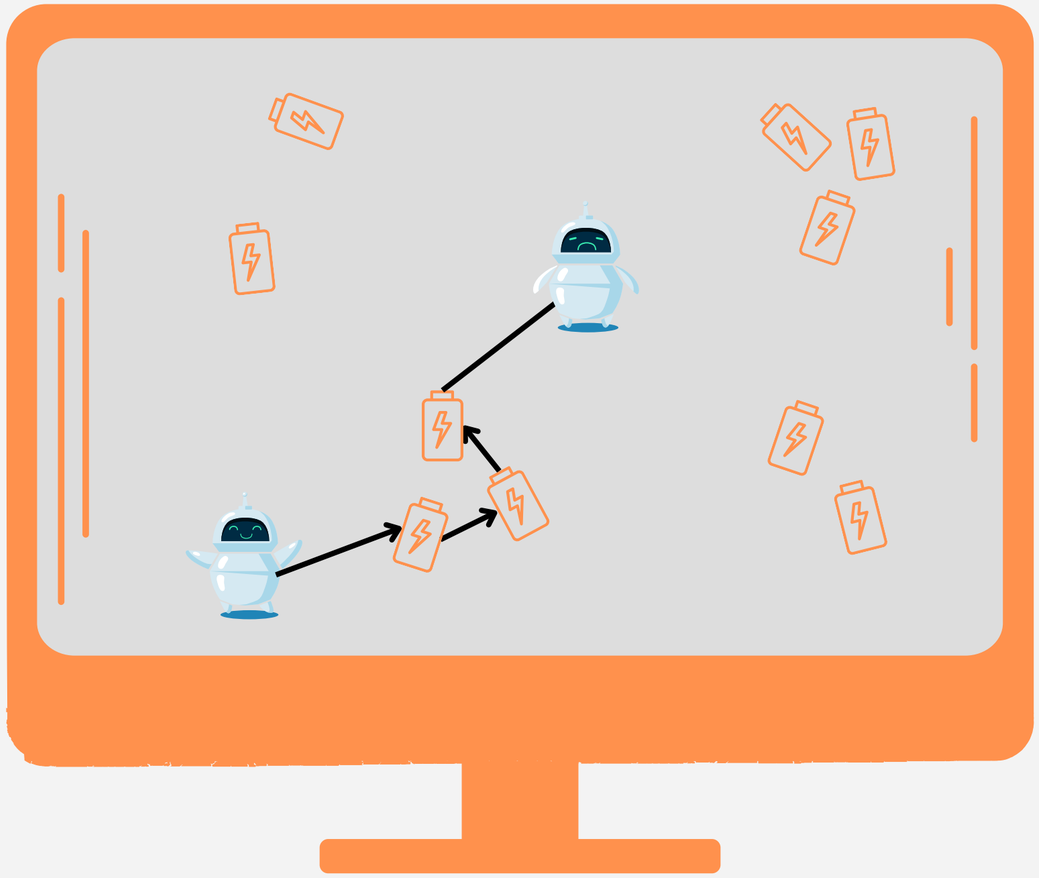
III Poission Disk Sampling

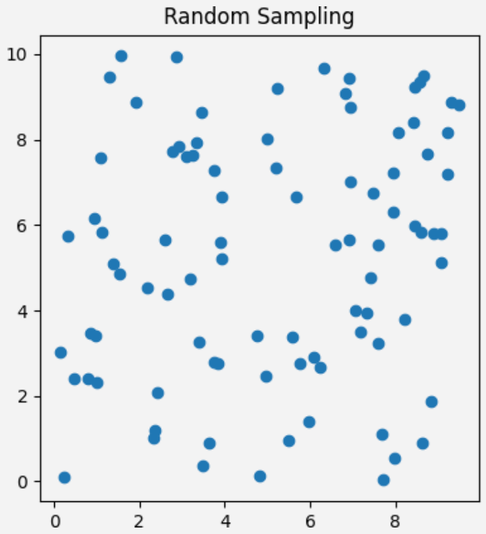


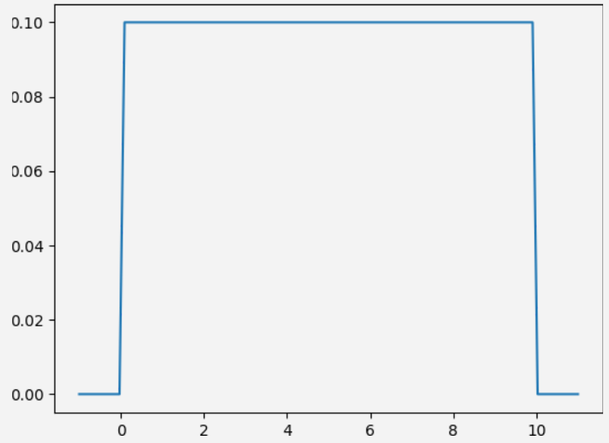
Task

- Collect Battery == good
- More collected Batteries == more good

III Poission Disk Sampling





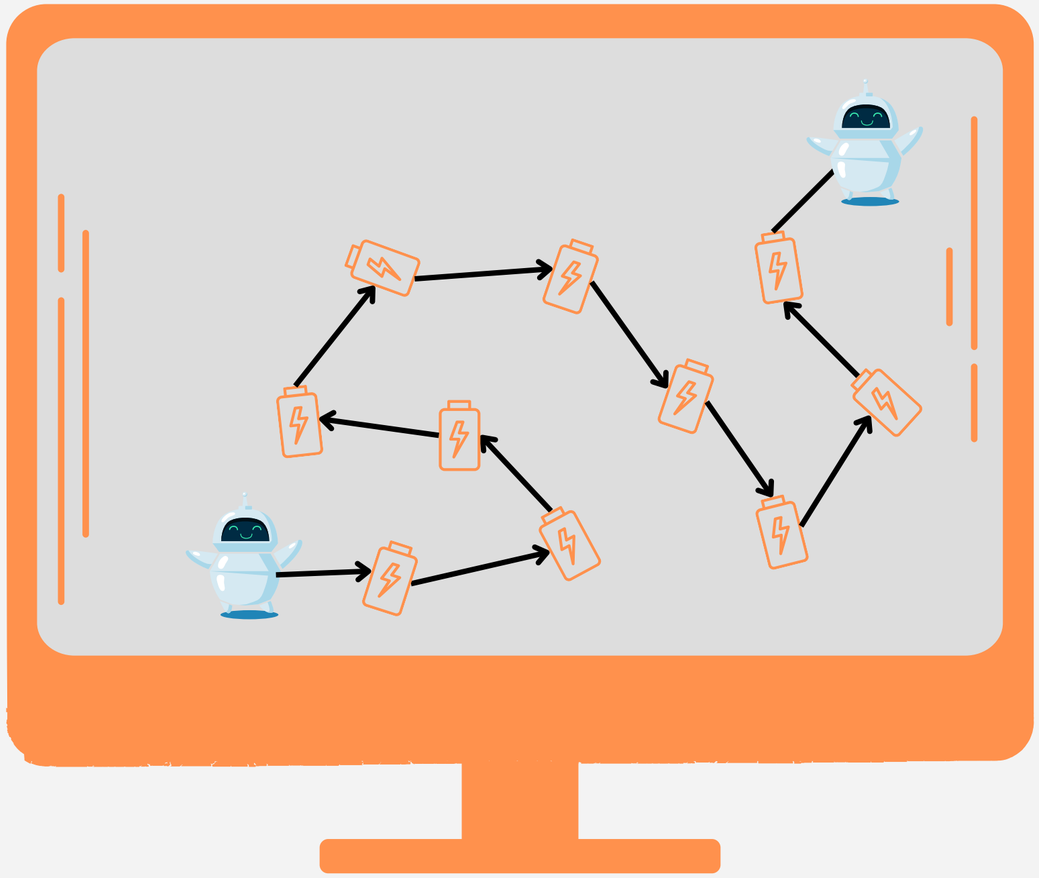


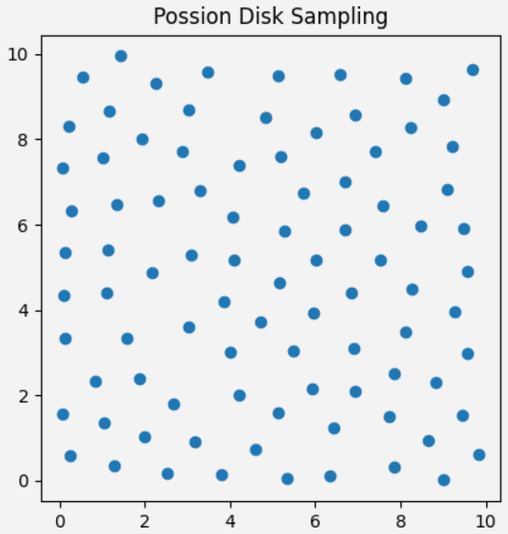
Problem

Clusting, can't score better ratings

if cannot leave local cluster

III Poission Disk Sampling



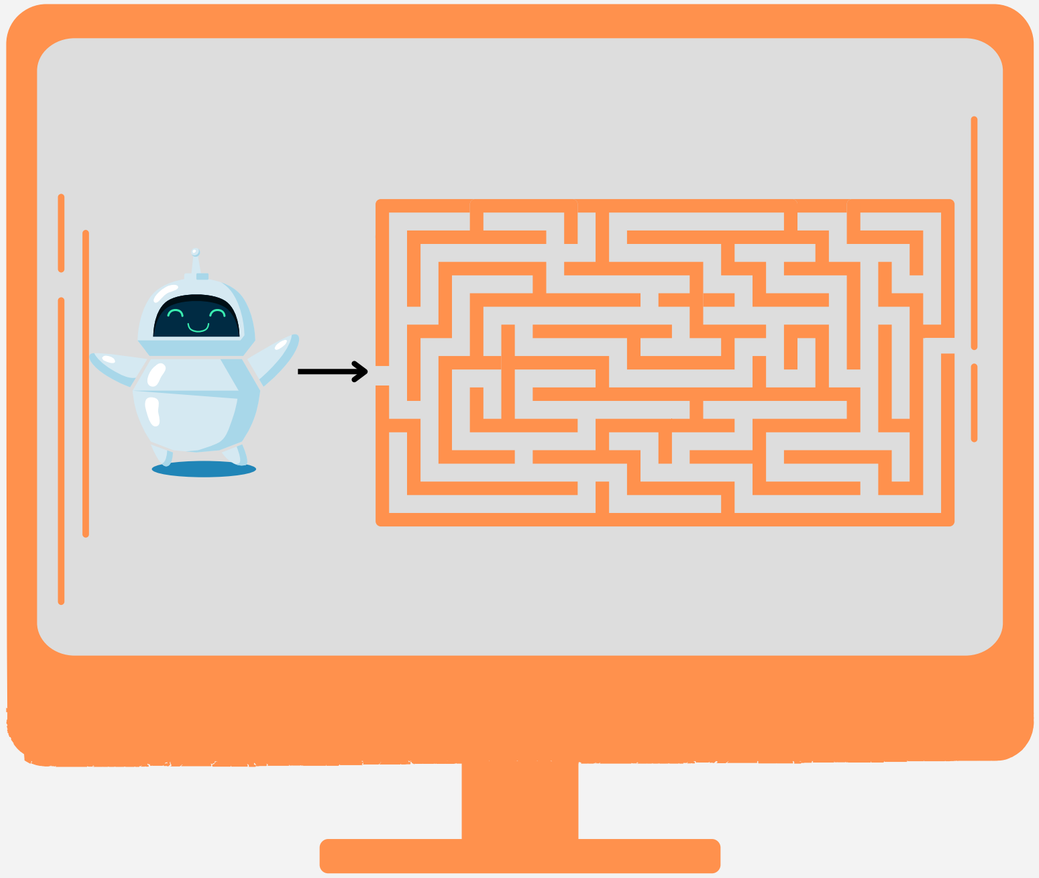


Properties

- Equally spaced samples
- aka Blue Noise

Wave Function Collapse

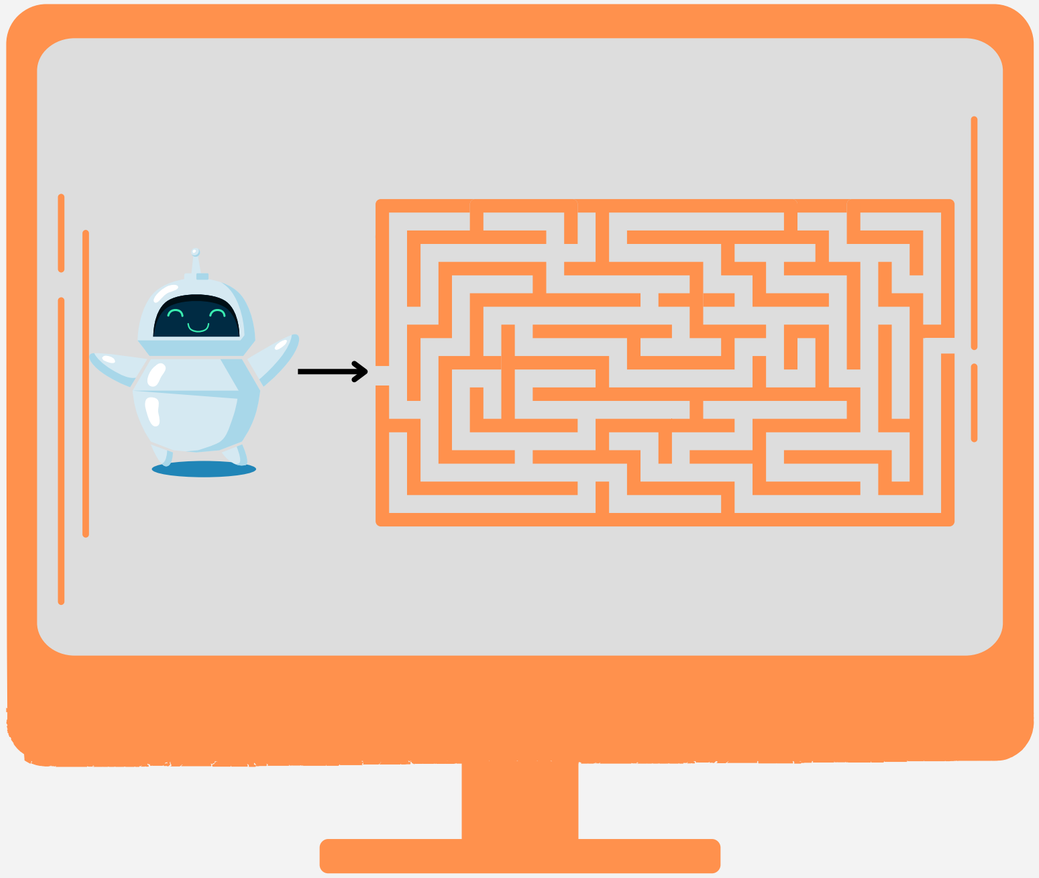
IV Wave Function Collapse



Task

Find the exit of a maze

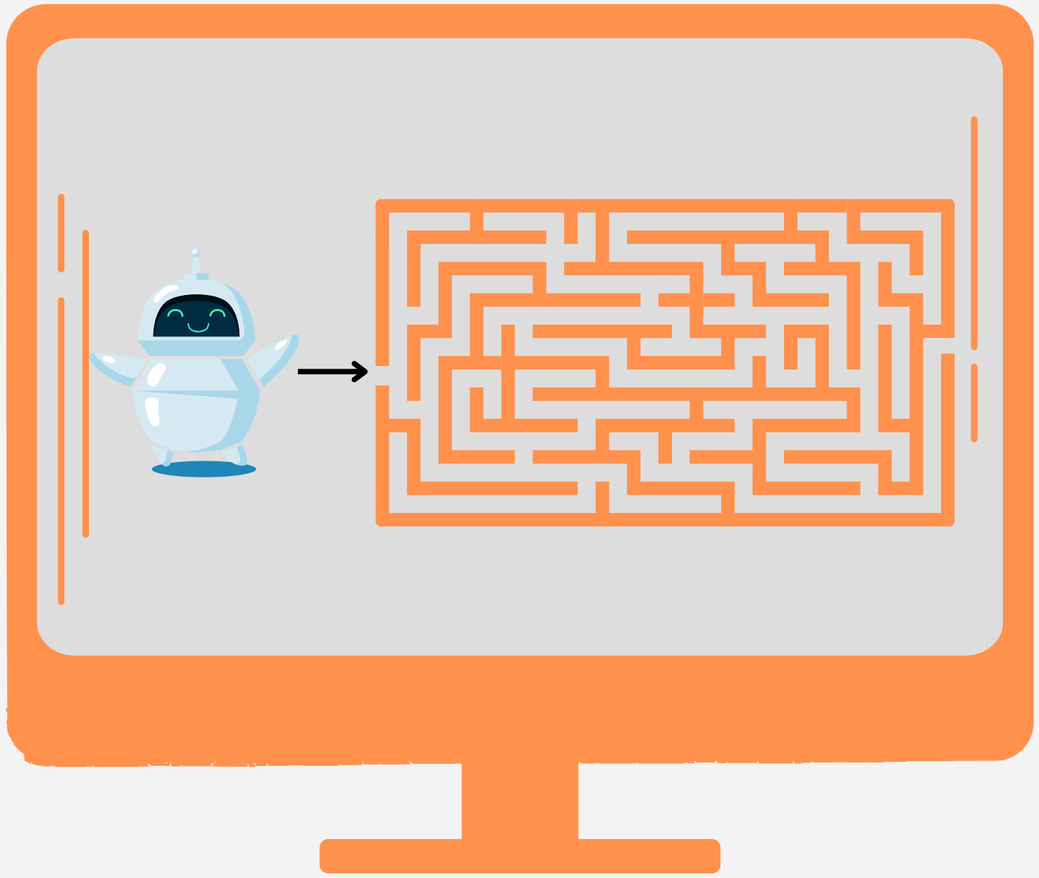
IV Wave Function Collapse



Problem

Generate mazes based on an example

IV Wave Function Collapse



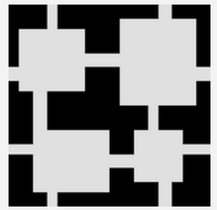


Image adapted from [3]

Properties

- Output locally similar to input
- n-dimensional

IV Wave Function Collapse

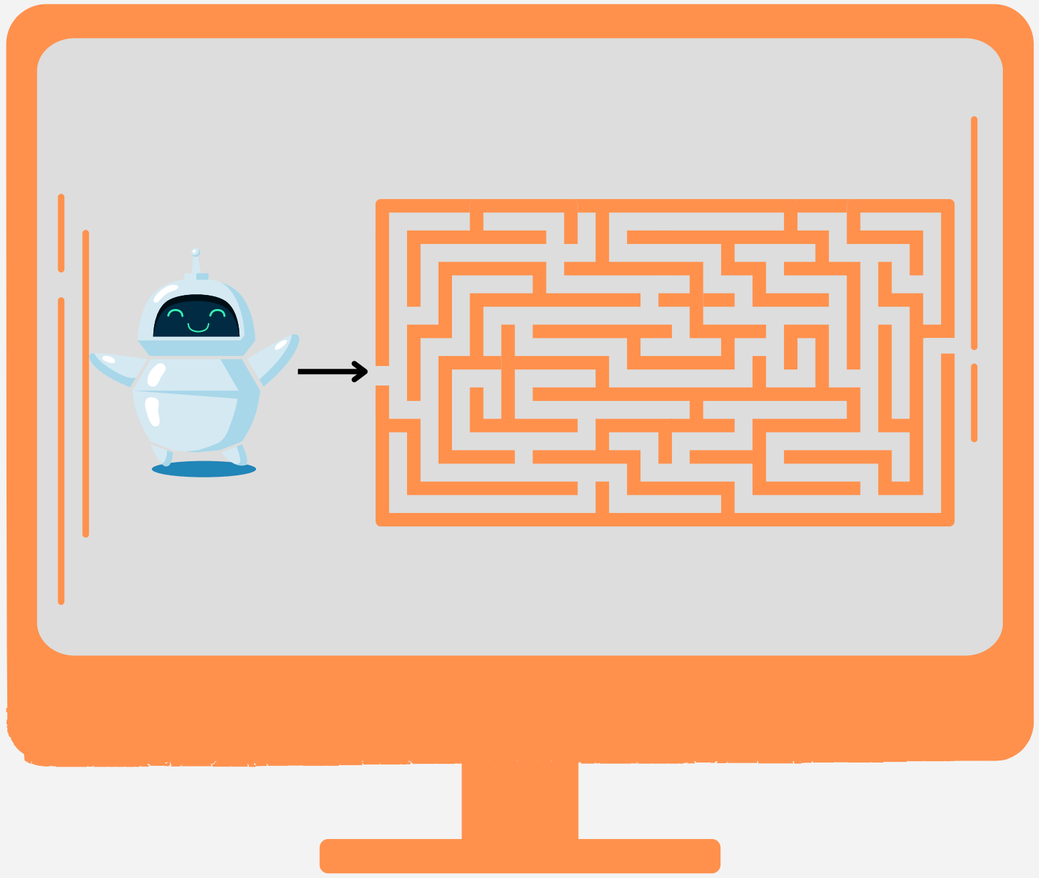


Image adapted from [3]

Properties

- Output locally similar to input
- n-dimensional

IV Wave Function Collapse

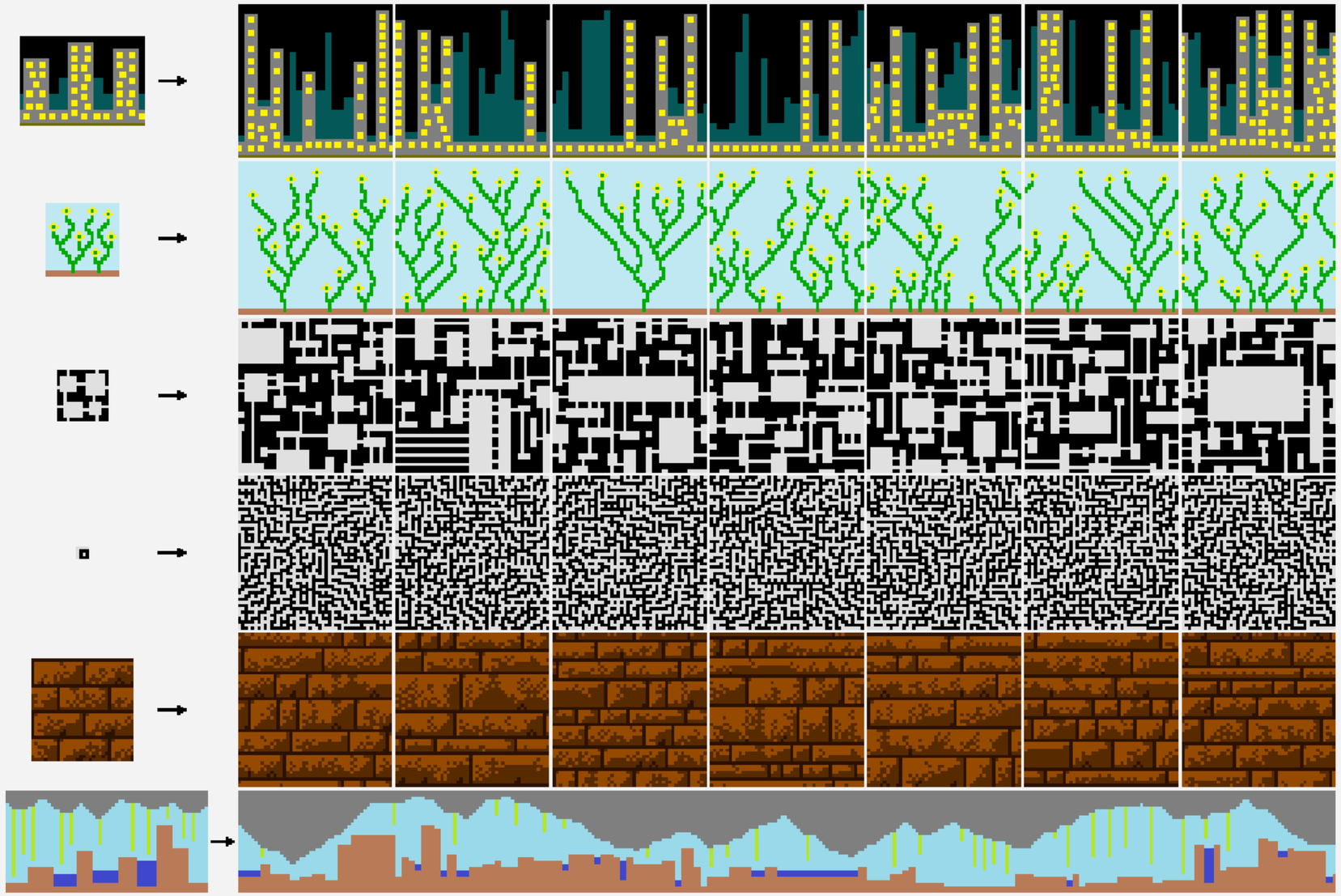
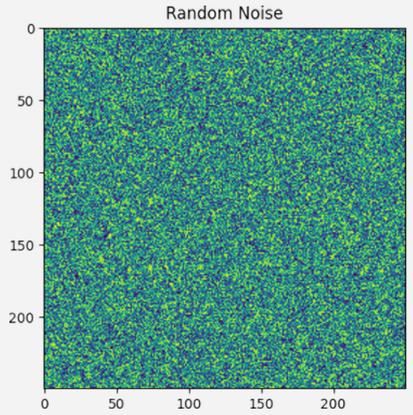


Image adapted from [3]

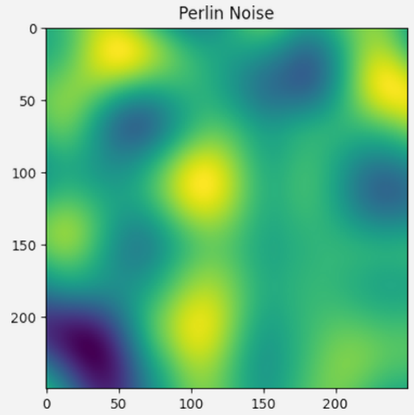
Wrap Up

V Wrap Up



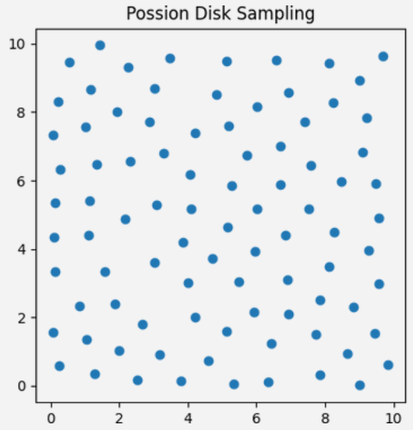
Random Noise

- uncorrelated
- downsides as model for sensor noise



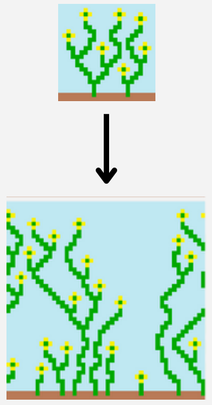
Perlin Noise

- Correlated
- Great for procedural Generation



Poission Disk Sampling

- Aim for certain density
- prevent clustering



Wave Function Collapse

- Generate more from less
- Can be tricky to tune

V Further Reading

1. Perlin, Ken (1985). An Image Synthesizer. SIGGRAPH 19.
2. Bridson, Robert (2007). Fast Poisson disk sampling in arbitrary dimensions. SIGGRAPH 07.
3. Wave Function Collapse Example Repository on Github
4. Herbert Wolverson's talk on "Procedural Map Generation Techniques" on Youtube

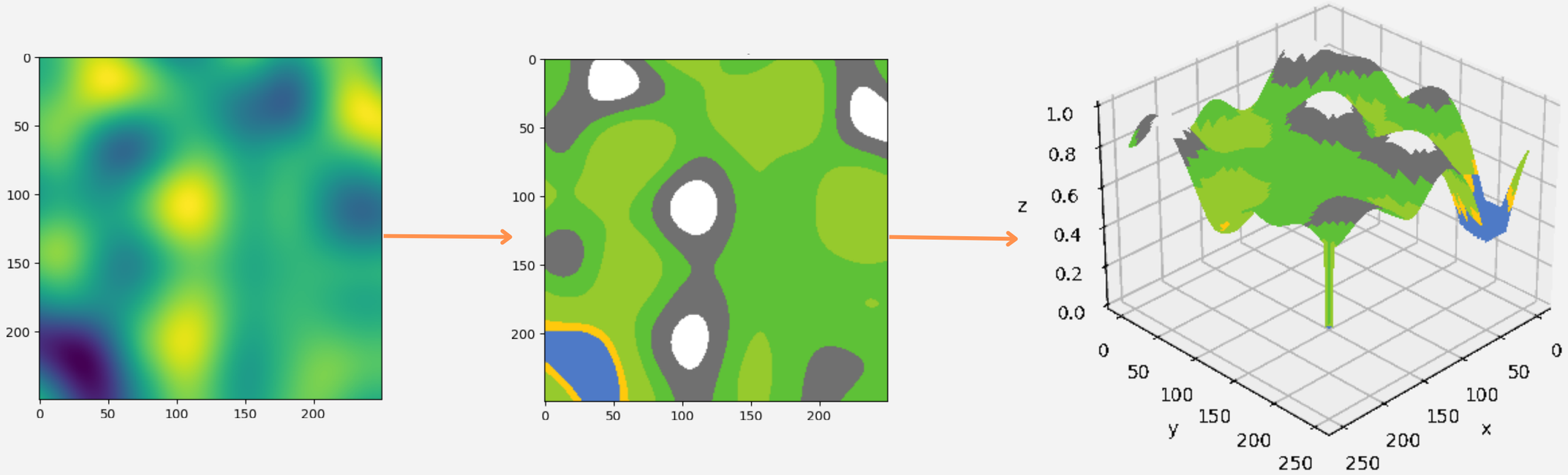
Thank you for listening!

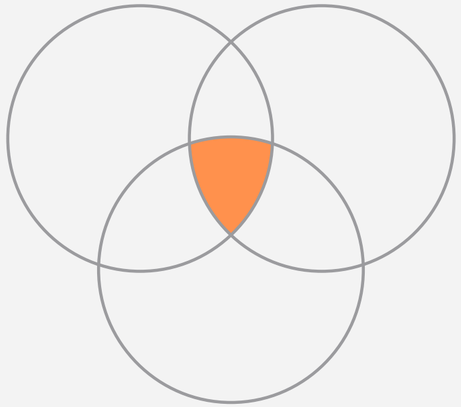
Sedetius

sedetius.com/nook22

[QR-Code, not noise]

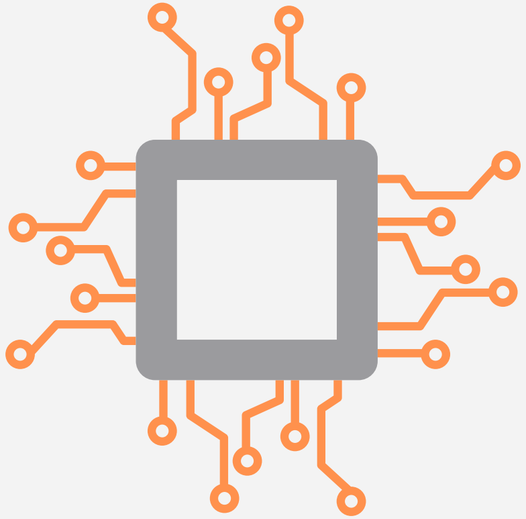
II Perlin Noise





Layer Noises

Combine different noises for different purposes. Layer noises of the same type to create fractal noise.



Resource Consumption

Some implementations require wast amounts of memory. Use a single, static instance for the whole program.

Safety

For safety critical systems noise might not be good enough. In these cases you need to use real sensor data.