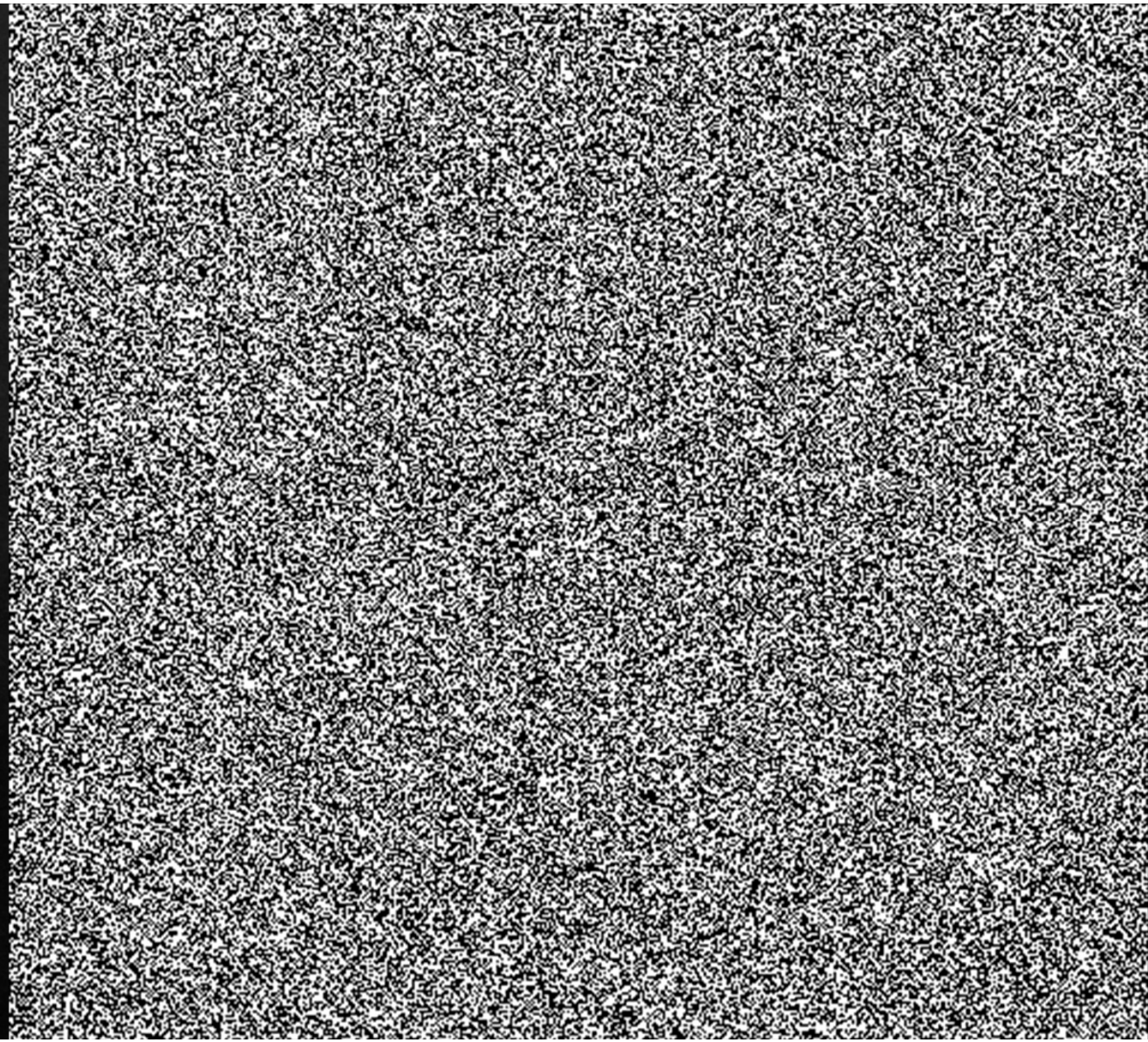

Natural noise generator using CUDA
Sebastian Resendiz
A01701111

Context

- Randomness is everywhere in computing
- Can be created by physical, or mathematical means
- Sometimes there's need to get more controlled randomness

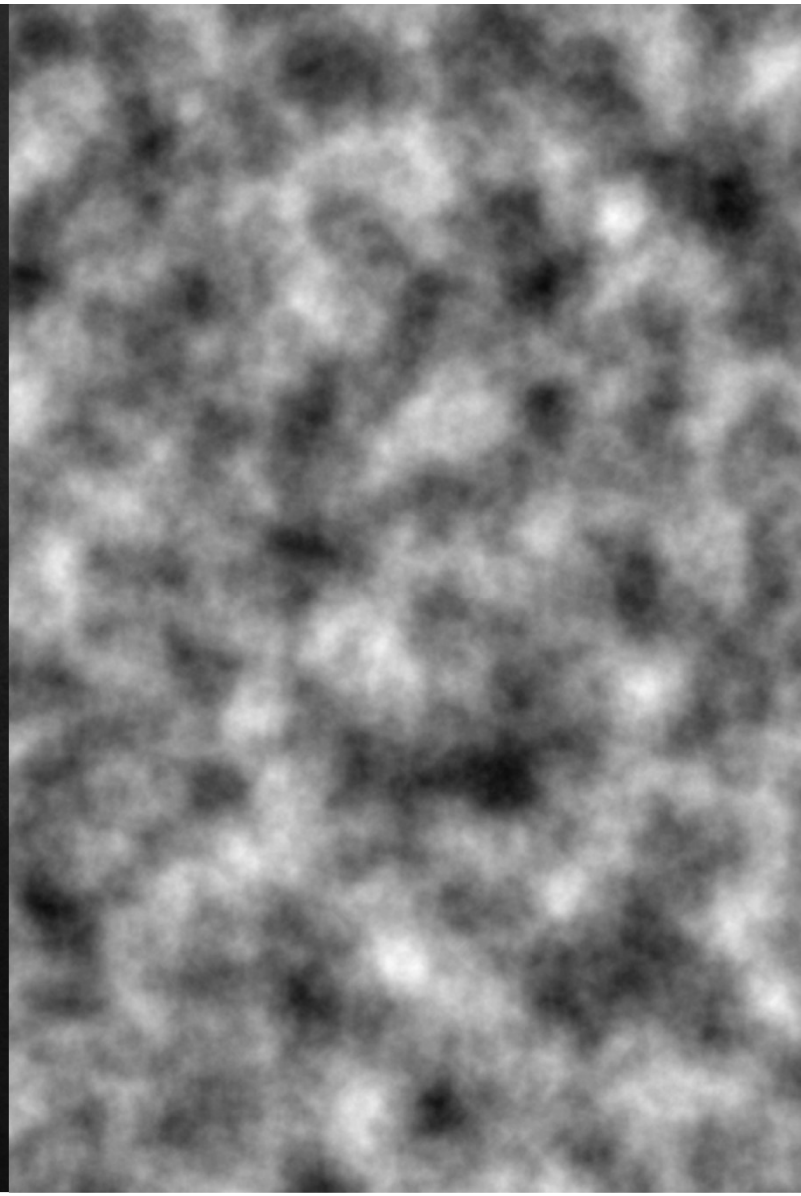
Problem

- Randomness looks ugly
- Its hard to get something natural looking



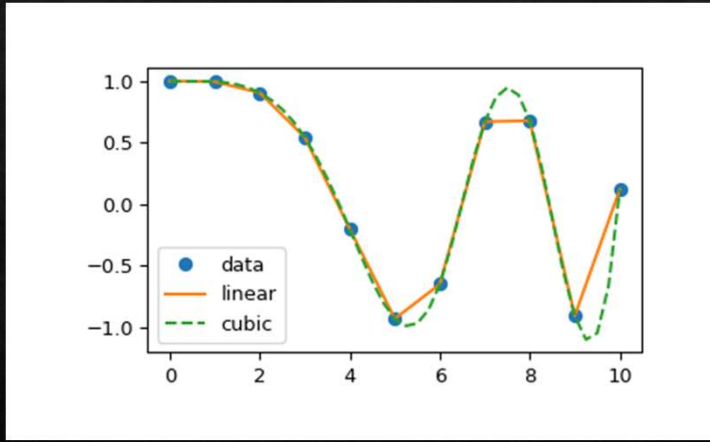
Solution

- Create a more organic randomness generation
- Use mathematics to compute the missing parts



Solution

- Get random or semi random numbers
- Smooth the numbers
- Interpolate
- Repeat by octave
- End!



```
-0.0966 0.0653  0.3682  0.3948  0.0328  -0.1832 0.1575  -0.2398 0.3380  0.1106
0.0800  0.3360  0.4082  0.0334  -0.2331 0.2168  -0.1601 0.3117  0.1897  -0.0484
0.3614  0.3387  -0.0883 -0.2226 0.1241  -0.2217 0.2481  0.0159  0.0390  0.0400
0.5143  -0.0628 -0.2590 0.1898  -0.2950 0.2769  0.0175  -0.0314 -0.0404 0.0434
-0.0281 -0.1712 0.1237  -0.2481 0.2311  0.1129  0.0527  0.0315  0.0499  -0.1315
-0.2198 0.1792  -0.2261 0.2292  0.1476  0.0025  -0.0797 0.0092  -0.2070 0.1124
0.2168  -0.1601 0.3117  0.1897  -0.0484 -0.0115 0.0529  -0.0673 0.1246  0.0981
-0.2217 0.2481  0.0159  0.0390  0.0400  0.1479  -0.1313 0.1107  0.2041  -0.0922
0.2769  0.0175  -0.0314 -0.0404 0.0434  -0.1490 0.1917  0.0818  -0.0419 0.4942
0.1129  0.0527  0.0315  0.0499  -0.1315 0.1436  0.0779  -0.0515 0.4536  -0.0628
```

Results

Conclusion

- It is useful to create more organic random numbers
- Its very handy
- Using CUDA does make it faster

Thanks!