Image Formation 3 & Contrast & Point processes

opencv: BGR
matplotlib: RGB

`img = [:, :, [2, 1, 0]]`
↑ color space transformation

3D array: R, C, color channel
[[[R,G,B], [R,G,B]]
[[R,G,B], [R,G,B]]]

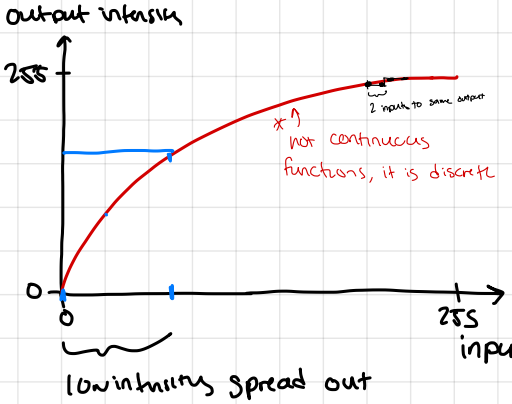
Problems in image formation:

Contrast: (over/under exposure)
resolution: (too small / too large image size)
Bayer artifacts: (color from demosaic)
Blur : exposure time, moving object
Aliasing : discretization
Noise : graininess

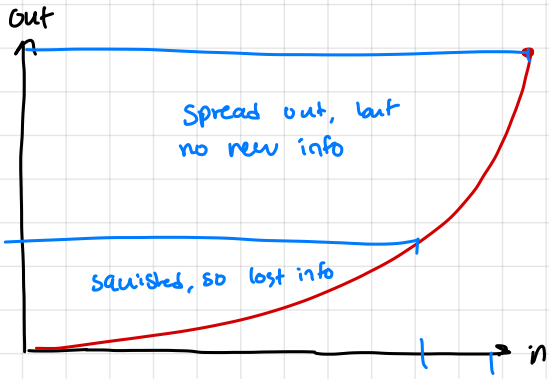
Contrast: difference between neighboring image regions
- tell where one thing ends & other one begins
- we would like to increase contrast btwn image regions

First Image processing algorithm

point process



- fixes underexposure
- increases intensity of low intensities
- push up & spread out low intensity



- fixes over-exposure
- spreads out high intensity

- no matter what, we lose information.

What if your image has underexposed parts & overexposed parts?

- apply ✓ to over exp & ⌜ to under exp part?

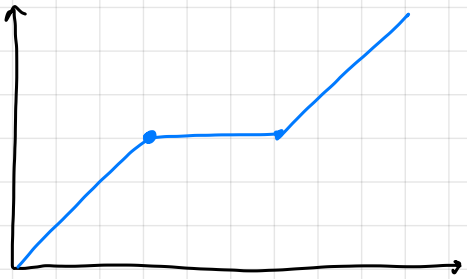
[UX, ... , 120, 121, ... , OV.EX.]

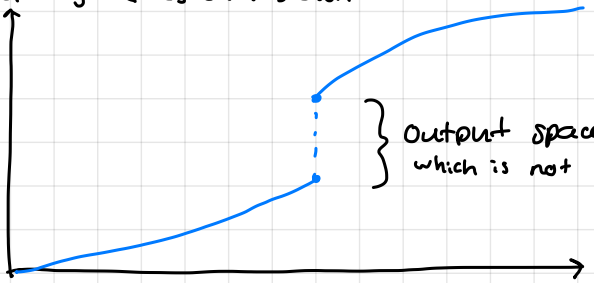
boundary issue flips intensity polarity

✗ bad

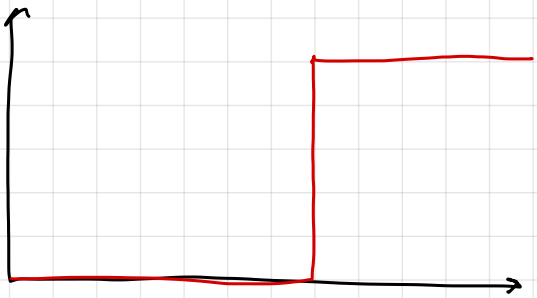
must be monotonic

- So we need a single point process for entire image & monotonically non-decreasing
  - ↳ this pixel only depends on its own value





- Thresholding point process



bright things appear, dark things disappear