Lecture 4

Digital Image Processing

Histograms and point operations

Say you have an Image and you want to improve its contrast. how does it work under the hood.

$I(x,y)$ ← pixel location

↖ gray scale value / colour → are discretized the grayscale and x, y

An inherit property of an Image is its histogram

define histogram as (The image)

$h(D) = \#$ of pixels in $I(x,y)$ that have Intensity D

$$= \left| \{ I(x,y) \mid I(x,y) = D \} \right|$$

The cardinality of a set is the measure of "number of elements of a set"

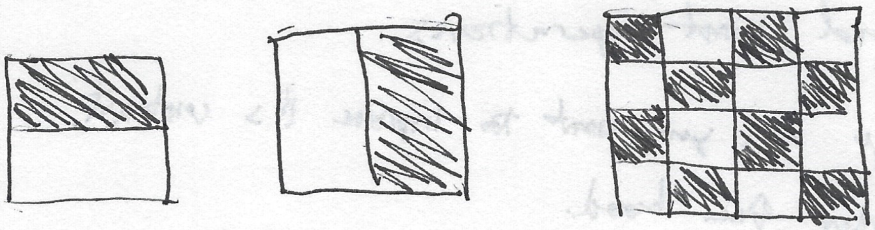
ex $A = \{2,4,6\}$, so $|A| = 3$ → "~~The ca~~ set A has cardinality of 3"

We may read it as the cardinality of the set of pixels where such that the pixel value is D (Intensity)

→ Matlab example around 3:24

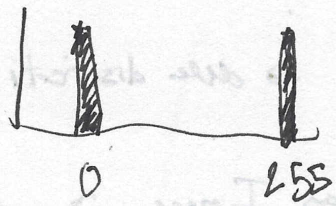
Note: histogram is not a 1 to 1 property of an Image. There are many Images that have the same histogram.

consider the images



The histogram for these images

will look like



we cannot confirm the image, but we can infer the pixel density distribution → good enough for contrast enhancement.

Point Operations

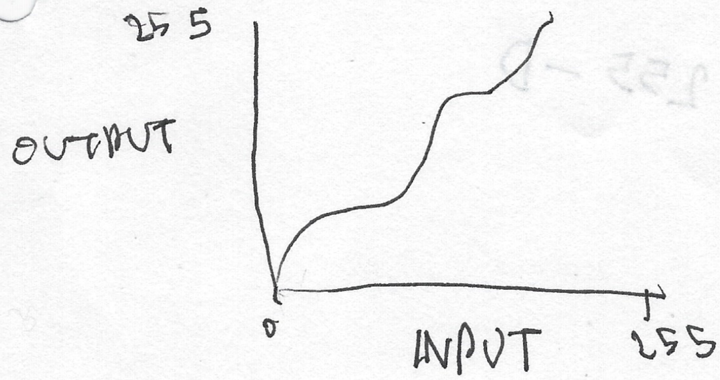
$$\underbrace{J(x,y)}_{\text{After}} = f(\underbrace{I(x,y)}_{\text{Before}})$$

Afterwards image

→ a functions of the image intensity

→ turn the intensity at x, y into a new intensity from D to f(D)

We want to map the ~~before~~ color to the after color

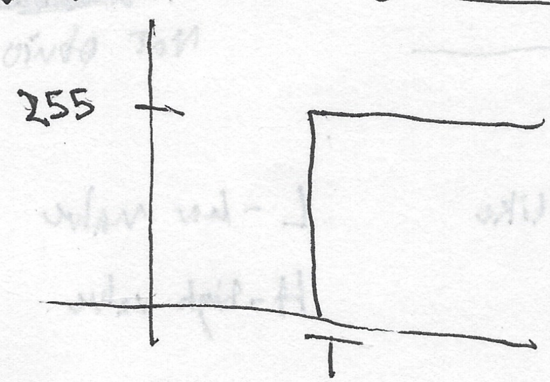


? changing the intensity of the pixels?

examples of these operations

→ Thresholding → happens a lot in Image processing and computer vision

$F(D) = H(D-T)\,(255)$ → scale it to 255

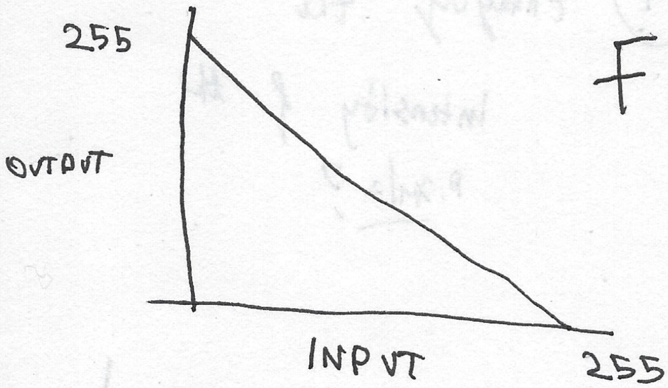


matlab example, before 9:00

J = (im > 200) → We lose information

→ This operation is not a 1 to 1 mapping

DIGITAL Negative



$$F(D) = 255 - D$$

→ matlab 10:10

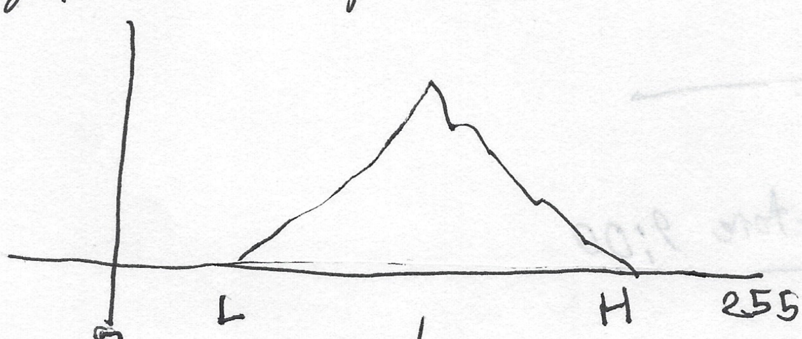
```
im = imread( ~ )
imshow(im)
imshow(255-im)   ← negative.
```

we are interested in the negatives as we may see certain information ~~to be seen~~ Not obvious

---

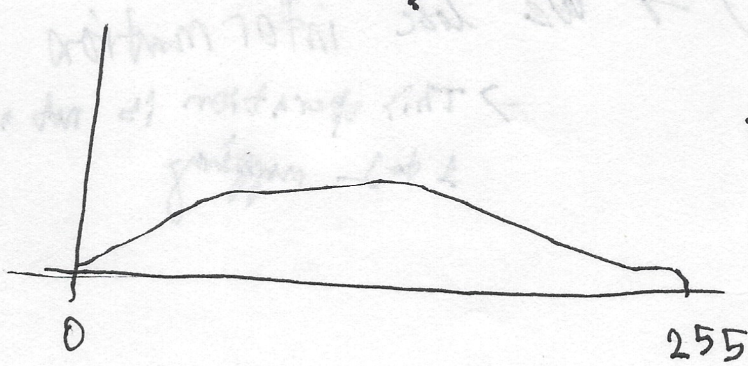
Contrast Stretching

suppose the histogram looks like



L - low value

H - high value



We scale the Lowest value to zero and the highest to 255

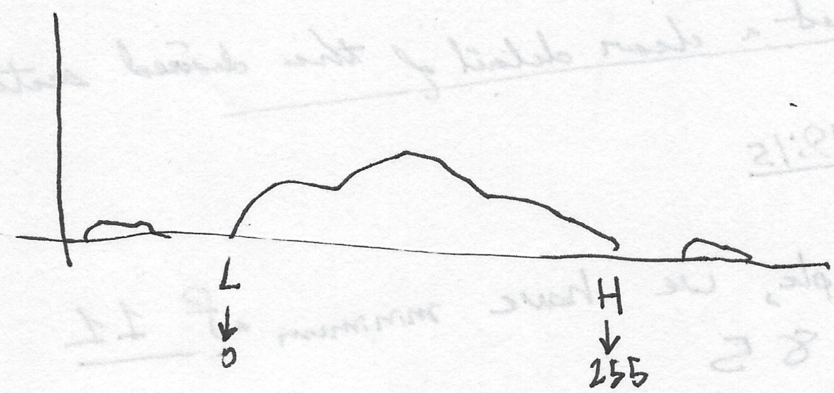
$L \to 0$ $\quad$ Assume everything else stretches

$H \to 255$ $\quad$ out linearly

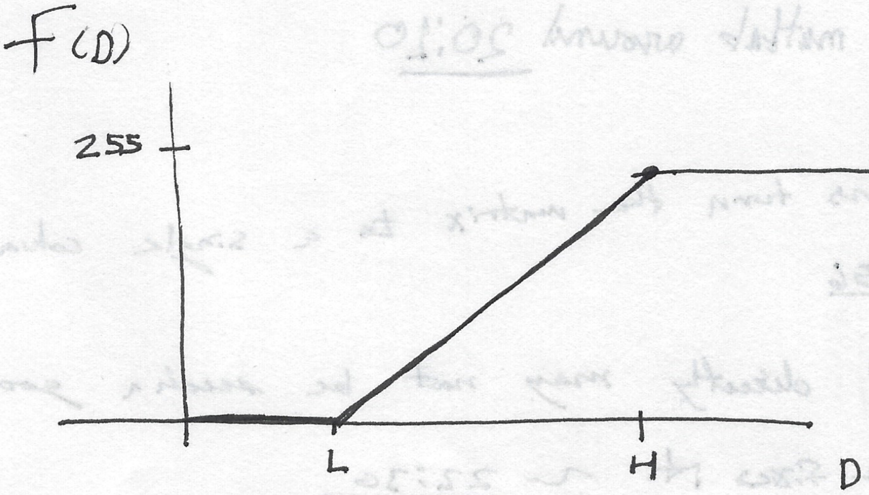
OUR function may be $F(D) = \frac{255}{H-L}(D-L)$

we recall that D is intensity (the x-axis of the histogram)

Realistically, we may have something that looks like



Realistically, we should clip the intensities to [0, 255]



imtool(im) % imtool command around 14:44

// The grey stuff becomes black and the light
// grey becomes white

⑥ we may not lose information in the image, we may be able to see details we would not be able to notice otherwise

→ the medical terms may be: windowing + leveling of medical Images.

→ We may use this technique to analyze an unclear portion of the Image to get a clear detail of the desired section.

Take note of 19:15

In the matlab example, we have minimum of 11 with a maximum of 85

Applying equation in matlab around 20:10

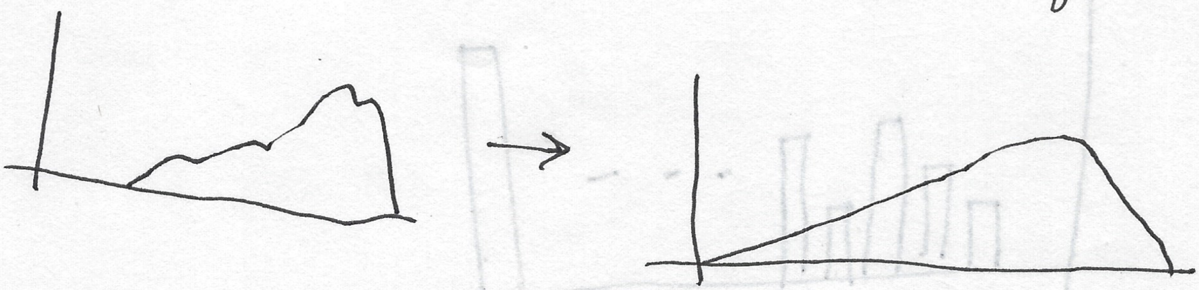
* note, V(:) means turn the matrix to a single column vector 21:56

* note using f(D) directly may not be such a good thing, so see how he fixes it ~ 22:30

imadjust 24:10

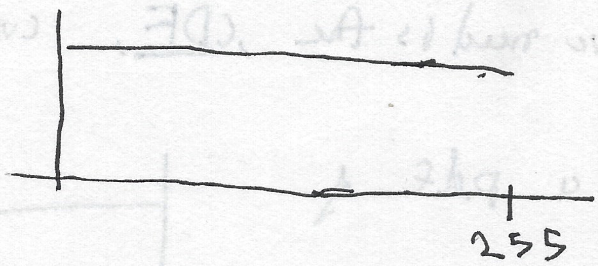
Histogram equilization 26:00

there are cases where contrast stretching doesn't work.



We may desire to create an image where there is an equal amount of the gray values as possible.

Desired Histogram



So what is F(D)?

Histogram is similar to a probability ~~distribution~~ density function. Histogram is a pixel count for every value of intensity (adding all values in the histogram should give the number of pixels in the image)

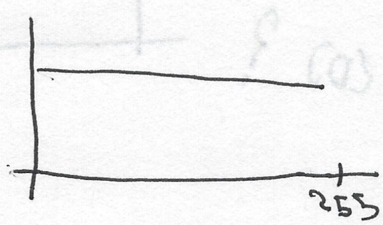
* Get the numbers of pixels, divide the values in the histogram by that number, you get a Probability Mass Function.

(8) Think of the Image histogram as a Probability Mass Function.

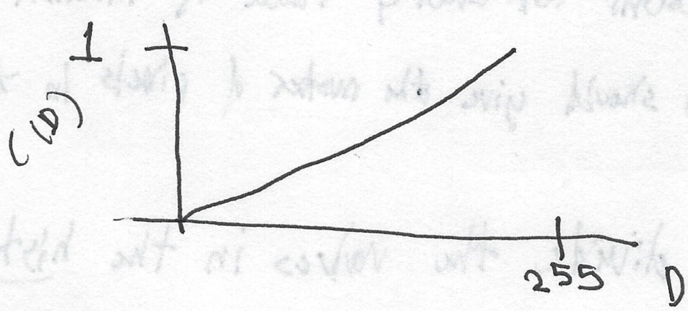
$$\sum_{D=0}^{255} p(I(x,y)=D) = 1$$

what we need is the CDF, cumulative Density function

With a p.d.f. of



we have a c.d.f. of



So the Idea is that we want to turn our old c.d.f. into the new cdf.

$$C(D) = P(I(x,y) \leq D)$$

So what mapping do we want?

→ what $f(D)$ converts the original image histogram INTO UNIFORM DISTRIBUTION?

Radke: turns out that function is the original CDF

So $F(D) = C(D)$ ← CDF of Original Image

STOPPED at 31:36

~~by~~ the functions goes from

$$I(x,y) \rightarrow J(x,y)$$

$$P(J(x,y) \leq D) = P(C(I(x,y)) \leq D)$$

$$= P(I(x,y) \leq C^{-1}(D))$$

$$= C(C^{-1}(D))$$

$$= D$$ where the Intensities are scaled from zero to 1 [0,1]

IN Practice

$$f(D) = \frac{\sum_{x=0}^{D} h(D)}{\sum_{x=0}^{255} h(D)} \rightarrow \text{scale to } [0, 255]$$

sum of the values of the histogram

all of the values of the histogram

→ with this implementation, we can't get the histogram totally flat

→ Matlab example → 34:42

Histogram SPECIFICATION

old Hist $h_1(D) \longrightarrow h_2(D)$ New Hist

$h_1(x) \rightarrow$ $\boxed{\text{Histogram equalization}}$ $\longrightarrow$ Uniform (Image) $\longrightarrow$ $\boxed{\text{Inverse Hist. EQ.}}$ $\rightarrow h_2(X)$

$$f(D) = C_2^{-1}\left(C_1(D)\right)$$

CDFs of Images

"Gamma Correction"

Problem: Every Display Device has a Different nonlinear relationship between Pixel Input Intensity and Display output Luminance

ex, the relationship between the current and the voltage was not linear in a cathode-Ray-tube

→ Images appear fine on screen, but on a projector, they would look washed-out. (Too dark)

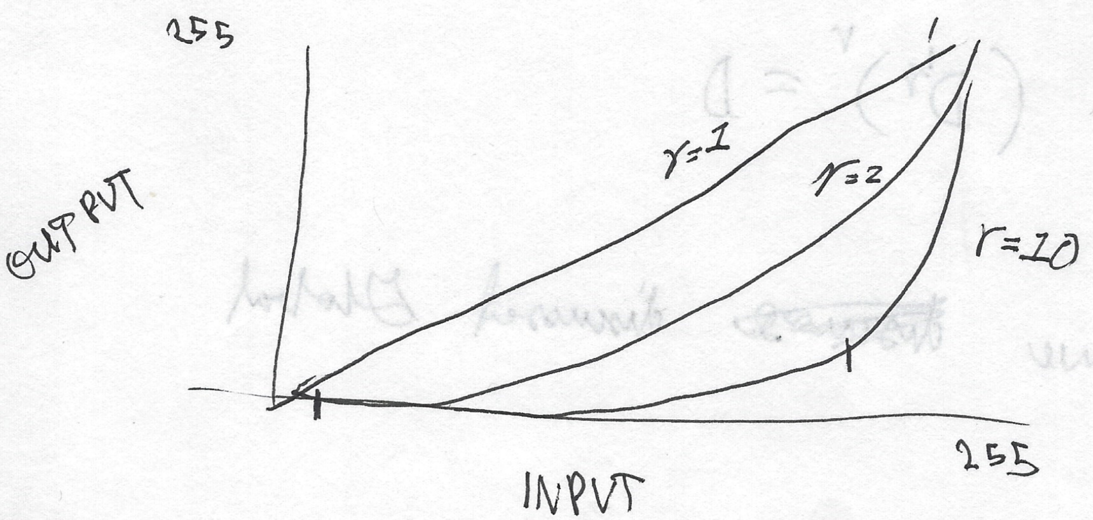
→ The mapping of pixel colors to my screen was a different function than the color pixels (mapping) that get sent to the projector.

This is Gamma Correction

The Relationship for real Devices is often modeled as a power functions

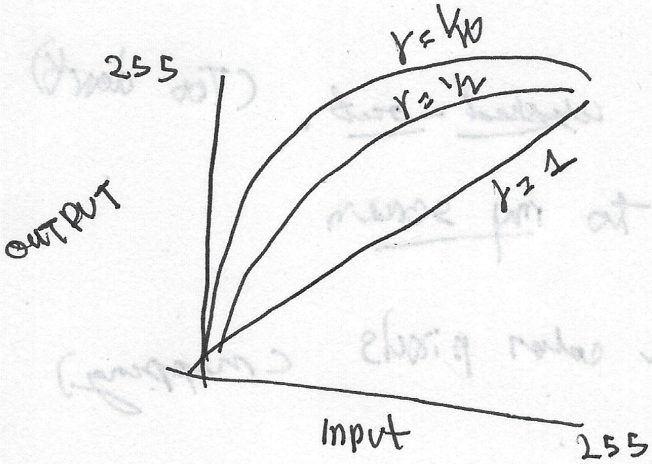
$$f(D) = D^{\gamma}$$

what do these functions look like?



So say that $\gamma = 10$, so $1 \to 150$ is getting mapped to a small window of the output.

(12) for $\gamma > 1$



Radke claims this issue has been mitigated

→ he's not an expert

VGA may cause issues

older versus newer

How to deal with this problem?

Solution: If we know the γ of the display device, → Precompensate Intensities.

we send Intensity ~~$(D^{1/\gamma})^{\gamma}$~~ $D^{1/\gamma}$ to the Display

That will be corrupted by the Gamma of the display to get the pixel we originally wanted

$$\left(D^{\frac{1}{\gamma}}\right)^{\gamma} = D$$

So far we ~~discuss~~ discussed Global operations, Point OPERations Affect every pixel

the same grayscale value the same way
(with the) (Intensity) / in)

→ Say we want to do local contrast enhancement or
local histogram equilization.

Many Image Process operations are more local.

one possibility

eg. Spatial Filters, So DSP is a pre-requisite because these spatial filters are analog to the FIR filters

e.g. Filter an Image to replace each pixel
by the average of ~~its~~ its neighbors

$$J(x,y) = 1/9 \sum_{i=-1}^{1} \sum_{j=-1}^{1} I(x-i, y-j)$$

Avg of the 3 pixels to the left, Right, up, down

$$= \frac{1}{9}\left[I(x,y) + I(x-1,y) + I(x+1,y) + \ldots \right)$$

$$\begin{array}{|c|c|c|} \hline 1/9 & 1/9 & 1/9 \\ \hline 1/9 & 1/9 & 1/9 \\ \hline 1/9 & 1/9 & 1/9 \\ \hline \end{array} = \frac{1}{9} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

(14) Matlab, 52:26

→ we make the Image blurrier

why would we want to do this?
→ Get rid of noise

denote $\frac{1}{9}\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix}$ as $h(x,y)$

We may write,

$$J[x,y] = \sum_i \sum_j h[i,j]\, I[x-i, y-j]$$

Looks scary, looks like convolution

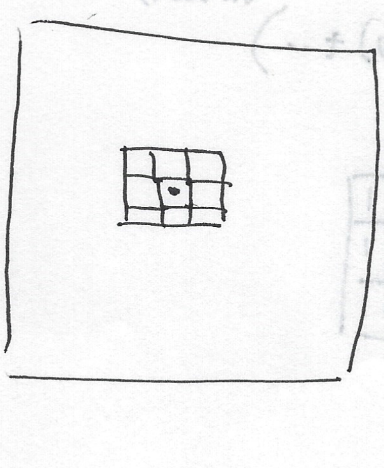
2D convolution of this 2D Spacial filter with ~~the~~ whole Image

As stated

$$J[x,y] = h * I$$

Convolution of filter and Image

You are familiar with 1D convolution



pixel filter over a center pixel, apply a combination of the ~~pixel~~ values of the pixels surrounding the center pixel to get our result

→ the result is a single pixel

Something more unexpected

examine

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

→ middle pixel is multiplied by 8, and subtract all of the neighbors

→ for a block of uniform color, <u>nothing will happen</u>. look at <u>58:18</u>

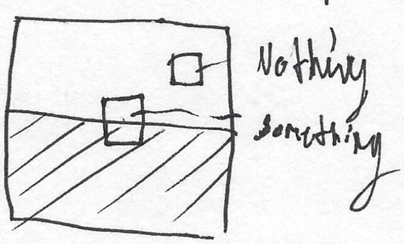
Spatially selective filter.

→ EDGE-DETECTING SPATIAL Filter

Suppose

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

this responds to sharp edges (Horizontal)



→ Matlab, <u>1 hr</u>

Next lecture we change the shape of the pixels