EE262
Handout # 2

Two-Dimensional Imaging
(class web site: www.stanford.edu/class/ee262)

Class overview and procedures

General info in handout # 1, and on web

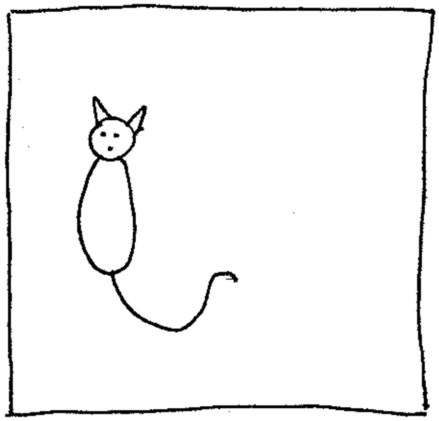
Class materials
- Text - Bracewell, Fourier Analysis and Imaging
- Additional books on reserve in Terman Eng. Library
- Handouts - usually every lecture, will supplement material in text
  - Handouts on class web site www.stanford.edu/class/ee262
- Computer - can use any machine you like
  - disk space augmented on Leland system
  - access to SCIEN lab
  - can program in any language you like
    C, Fortran, Matlab, ...
  - will have to be able to read data files

Grading
- Homework 30-35%
- Midterm 25%
- Final project 40%
- Other/extra credit 5%

Backgrounds -
261, computer familiarity at least at Matlab level

Properties of Images

What do we mean by "Images" and "Imaging"?
Consider an "image" of a cat:



What do the following have in common, and how do they differ?

A photograph of a cat
A tv image of the cat
A picture on a friend's home page of his cat
An MRI scan of a cat

We will define here an image as a two (or three) dimensional distribution of some quantity that can be, and usually is, displayed to be sensed by the human eye. In this class we'll consider both mathematical and computational tools for forming and manipulating these data.

Imaging as a system or machine entails

i) design of imaging systems
ii) image acquisition

iii) image processing
iv) image analysis
v) image interpretation

There are many imaging systems of which you are aware:

Optical Systems
X-ray systems
Tomography
Sonar
Radar
Acoustic imaging, esp. for medicine
Seismic Imaging

All involve source distributions, propagation effects, detectors, analysis algorithms, and displays. Traditionally interpretation is done by trained (or untrained) human operators, but many automated approaches are being developed.

Can you think of applications for ~~aomat~~ automated image interpretation?

<u>Some notation</u>

Most of what we'll use in Bracewell text

$$\text{sinc } x = \frac{\sin \pi x}{\pi x} \qquad \text{jinc } x = \frac{J_1(\pi x)}{2x}$$

$$^2\text{sinc}(x,y) = \text{sinc } x \ \text{sinc } y$$

1-D Fourier transform $\quad f \supset F$

2-D Fourier transform $\quad f \stackrel{2}{\supset} F$

Examples: $\text{sinc } x \supset \text{rect } u$

Number of dimensions not always obvious:

1-D: $\quad \delta(x) \supset 1$

2-D: $\quad \delta(x) \stackrel{2}{\supset} \delta(v)$

In case of confusion we can more clearly state the latter as

$${}^2\delta(x) \stackrel{2}{\supset} {}^2\delta(v)$$

Some functions inherently two-dimensional

$$\text{rect}(r) \stackrel{2}{\supset} \text{jinc}(q)$$

Here use of r, q as variables also clues us. (Recall $\text{rect } x \supset \text{sinc } u$)

Convolution:

Asterisk for 1-D $\quad f * g$

Double asterisk for 2-D $\quad f ** g$

Methods of representation

We have said that an image is a two-d distribution of a quantity, such as brightness. Represent the image as a function of two variables $f(x,y)$ (we'll use Cartesian coordinates for now).

For a photograph, $f(x,y)$ varies from complete reflectance ($f=1$) to completely black ($f=0$).

How white is white?

paper ~ 0.8

Snow ~ 0.9 (very white snow)

Black paper: $\sim 0.1 - 0.2$ (sometimes less)

The moon has an average ~~att~~ albedo (reflectance) of just under 10%, hence it would appear black if not surrounded by very dark ($f \approx 0$) space!

We call the range of reflectances the contrast ratio or dynamic range of an image. Thus for photographs a range of $0.8/0.1$ or 8:1 is attainable. As EE's we express ratios as dB, 8:1 gives a 9 dB dynamic range.

In principle, the 8:1 or 9 dB of dynamic range can be divided up any way we please, so that a photo may be used to convey an unlimited amount of information. But we know that all real signals contain "noise", and that will limit the number of independent levels we can express in an image.

For example, suppose that the noise or uncertainty in a brightness measurement from the above example is about 10% in reflectance. Then the total range of 10% to 80% results in 7 "grey levels" achievable in the image:

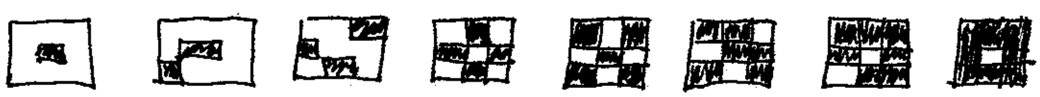
$$\frac{80\% - 10\%}{10\%} \quad \text{or} \quad \frac{\text{Max reflectance} - \text{min reflectance}}{\text{uncertainty in reflectance}}$$

→ How many gray levels in principle are representable using an 8-bit computer display?

Half-toning

A laser printer can only write white or black dots on a piece of paper. How then can we form images using them? The dot density and placement can be used to generate in-between shades of grey.

Example: 8 grey levels in a 3×3 box of dots

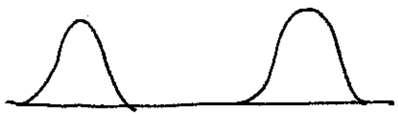


A random statistical method works also, with dot on probabilities set by reflectance values.

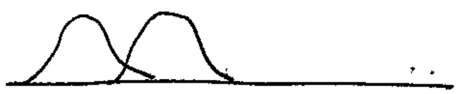
Magazines and newspapers vary the dot sizes in addition to the densities.

Resolution

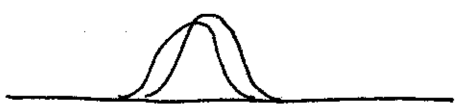
Another very important image property is resolution, or how fine an object may be distinguished in the image. We can think of resolution as how close two objects may be and yet be interpreted as separate entities. A 1-D example:



Two easily separable objects



Somewhat separable objects



Not easily separable

Often we use the width of an object at its half-power point (the 3-dB width since $-3\text{ dB} \approx \frac{1}{2}$) as a measure of resolution:

$\leftarrow$ point response, impulse response

Some resolutions of display systems:

Photographic emulsion:	1 μm grain size (binary)
Laser printer:	600-1200 dots/inch (40 μm)
Computer monitor:	0.25 mm dots size (8 bit brightness)

For optical systems we often use angular resolutions, where angular resolution $\Delta\theta$ is related to linear resolution Δl

by $\Delta\theta = \dfrac{\Delta l}{\rho}$, ρ is the imaging distance.

The human eye can distinguish $\Delta\theta \approx \frac{1}{3000}$ radian $\approx \frac{1}{50}$ degree

Atmosphere-limited telescope $\Delta\theta \approx 10^{-6}$ radian $\approx \frac{1}{20000}$ degree

Theoretical Palomar telescope $\Delta\theta \approx 10^{-7} m \approx \frac{1}{200000}$ degree

For an atmosphere-limited telescope, what can we see on the moon?

$$10^{-6} \text{ radian} \times 400000 \text{ km} = 400 \text{ m resolution}$$

How many "pixels" form the moon at this rate?

Area of moon: $\pi \cdot a^2 = \pi \cdot (1738 \text{ km})^2 = 9.5 \times 10^{12} \text{ m}^2$

Area of pixel: $\pi \cdot 200^2 = 1.26 \times 10^5 \text{ m}^2$

$$\# \text{ pixels} = \frac{9.5 \times 10^{12}}{1.26 \times 10^5} \approx 75{,}000{,}000 \text{ pixels}$$

A typical digital image is 1000x1000 or 10^6 pixels, so a full moon image would be quite large.

Resels vs. pixels

You are probably aware of the term pixel (picture element) to describe the point spacing in a digital image. How does this relate to the resolution element we have been discussing?

The resel size is the width of the impulse response, in units appropriate to the image.

The pixel size is how often the intensity (or other) distribution is sampled.

In well-designed systems, resel $\approx$ pixel so the storage is used efficiently without undersampling the distribution. More on this when we discuss aliasing later on.

<u>Raster displays, matrix displays</u>

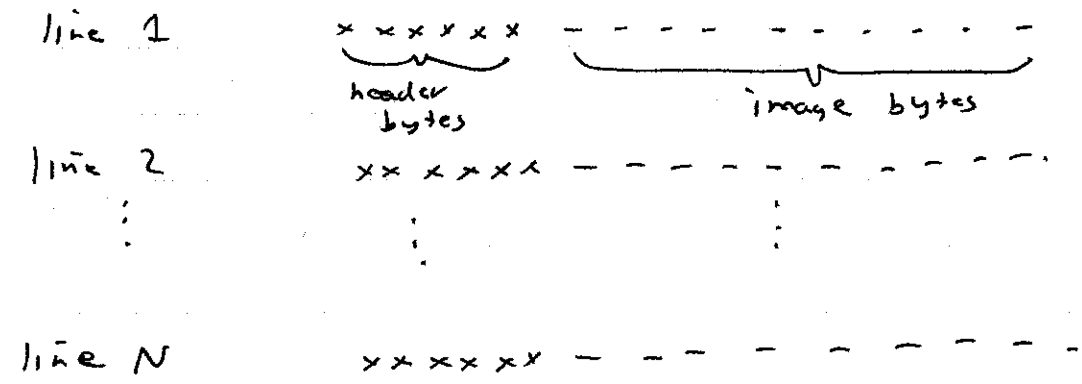
How can we store and display 2-D intensity distributions? First, assume we have quantized our function $f(x,y)$ into $f(x_i, y_i)$. By knowing the line length of an image we can go from an ordered list in a computer to a raster display.

Consider this sequence:

0 1 1 1 1 0 0 1 0 0 1 0 0 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 1 0
0 1 0 0 1 0 0 1 1 1 1 0

This "meaningless" string becomes an instantly recognizable picture when we know the line length and display it properly. If instead of ones and zeros, we use shades of grey we can create very complex and informative pictures.

Sometimes we (or others) include "header" information with each displayed line. This may, for example, include a "line count" so that lost data may be detected, or to ensure we can "line up" missing data.



<u>Contrast stretching</u>

Suppose we have a two-dimensional image function $f(x,y)$ we wish to display on a computer screen. How might we go about this?

We could simply create an array $a(i,j)$ to be displayed on the screen, and load each element i,j with quantized x_i, y_i values using the direct assignment

$$a(i,j) = f(x_i, y_i)$$

But what if the values of the function $f(x,y)$ are not well-matched to our display? Suppose that $a(i,j)$ is an array of bytes, each of course capable of being set to values ranging from 0 to 255. If f varies between 0 and 1, only the bottom two levels of the display will be exercised.

Clearly some sort of <u>scaling transformation</u> is needed to map f to a more efficiently, where efficiency here means optimal use of the levels of $a(i,j)$. We might model such an operation by a scaling function $s(\,)$:

$$a(i,j) = s(f(x_i, y_i))$$

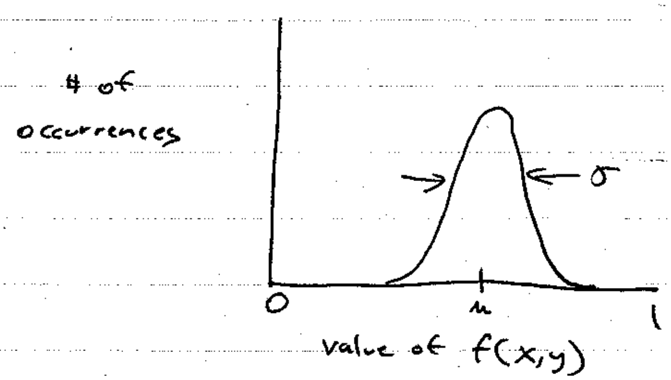
Lacking any other information about f, we could simply define

$$s(z) = 255 \cdot z$$

Then ranges of 0 – 1 in f yield 0–255 in a, and would display more pleasingly.

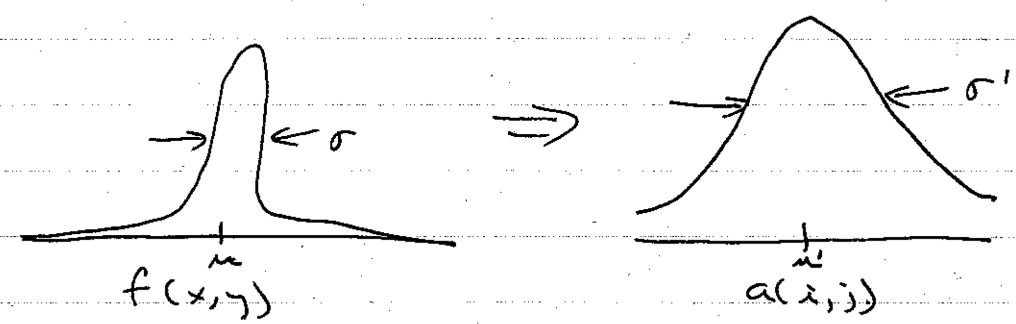
But the above ignore exactly how the range of f is distributed. Depending on how in the range 0-1 the values of f are mapped, we could optimize $s(\cdot)$ in some sense.

One way of assessing superior mappings is to examine a <u>histogram</u> of the input array. Suppose we create a histogram of an image $f(x,y)$ and obtain



This image has a rather narrow range of values centered on some mean μ. Mapping this into a via $S(z) = 256 \cdot z$ will still be wasteful of many potential levels in $a(i,j)$. Such an image will not have much contrast and will appear washed out.

A much better job is to transform values of $f(x,y)$ such that the histogram of $a(i,j)$ fills the available range:



Since σ' is greater than σ, the contrast will increase. Centering μ' in the range of $a(i,j)$ will assure us that the image is neither too light nor too dark. The optimal transformation $S(\ )$ is left to the reader to infer.