Image analysis

CS/CME/BioE/Biophys/BMI 279

Oct. 24, 2019

Adrian Sanborn

Ron Dror

Assignment 2 tips

- Your predictors will not perfectly recapitulate the structure of HRAS, because protein structure prediction is challenging.
- But you should get a fairly compact structure.
- TAs have added extra office hours next week.

Outline

- Images in molecular and cellular biology
- Reducing image noise
  - Mean and Gaussian filters
  - Frequency domain interpretation
  - Median filter
- Sharpening images
- Image description and classification
- Practical image processing tips

Images in molecular and cellular biology

Most of what we know about the structure of cells come from imaging

- Light microscopy, including fluorescence microscopy

Light-microscopy uses photons to excite a fluorescent protein or dye, which will in-turn emit light of a different wavelength, allowing for visualization of protein(s), cell(s), etc labeled with the fluorescent marker. Without these markers, we would have a very difficult time interpreting images or locations within an image

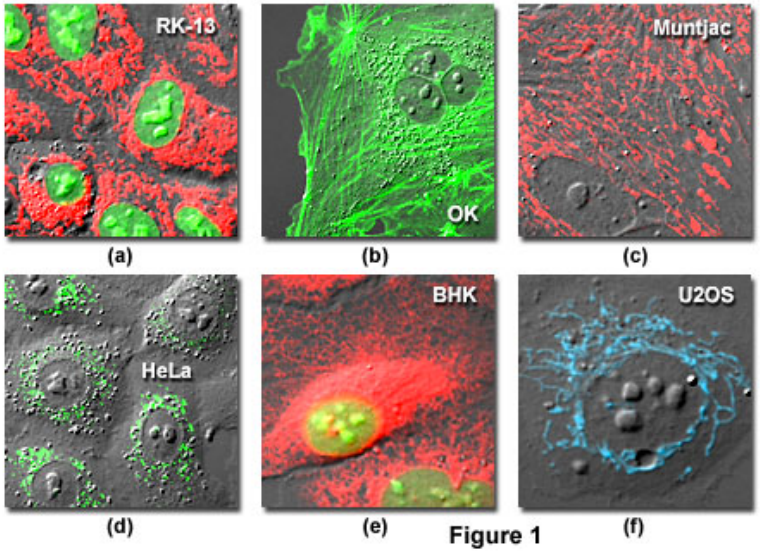


Figure 1

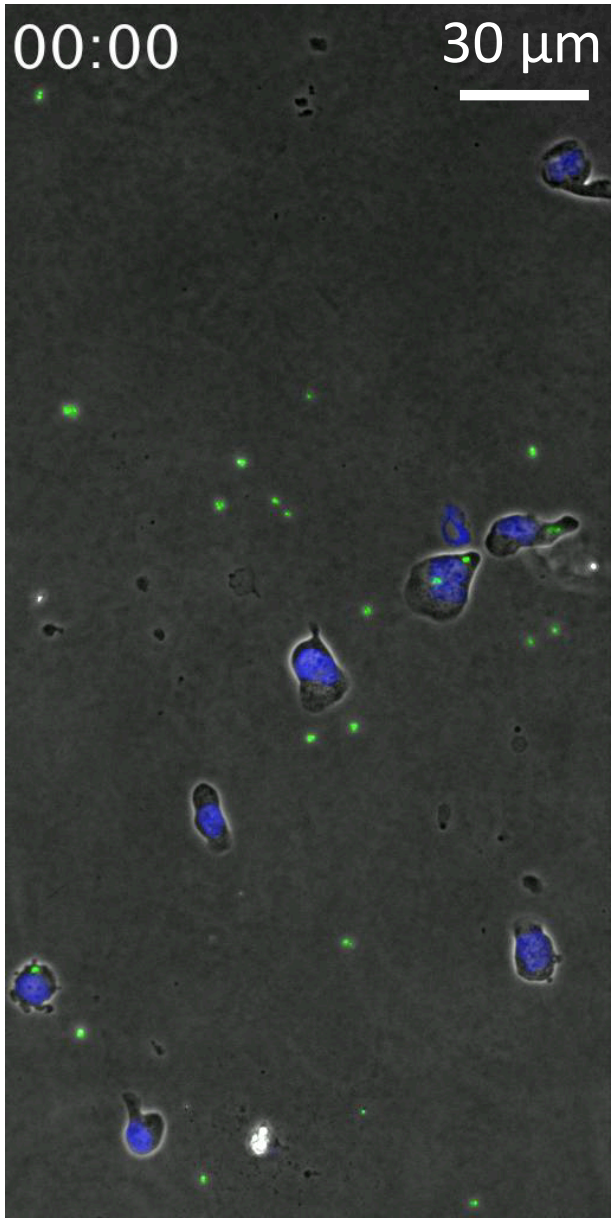
https://www.microscopyu.com/articles/livecellimaging/livecellmaintenance.html

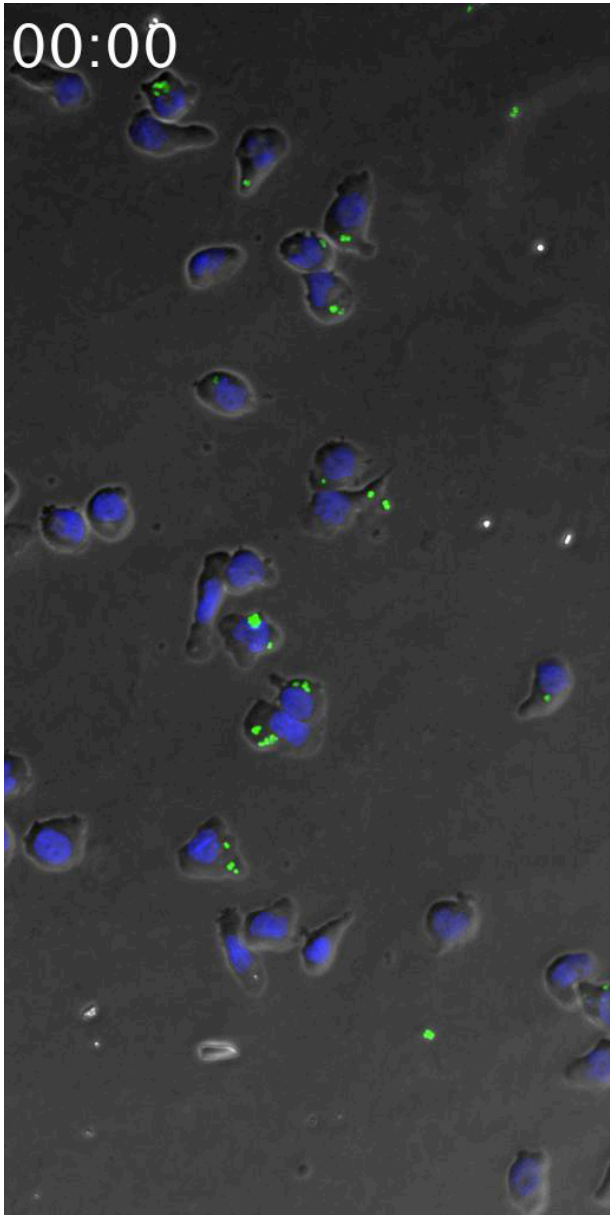
- Electron microscopy

Electron microscopy uses electrons to image a sample, providing much higher spatial resolution (on the scale of pm-nm) than light microscopy, but requiring specific conditions for the item being sampled; one such limitation is the requirement of the sample being in a vacuum, thus requiring samples to be dead.

http://blog.library.gsu.edu/wp-content/uploads/2010/11/mtdna.jpg

Imaging can capture structure over time





Human white blood cells eating bacteria

Nucleus

***S. aureus* (GFP)**

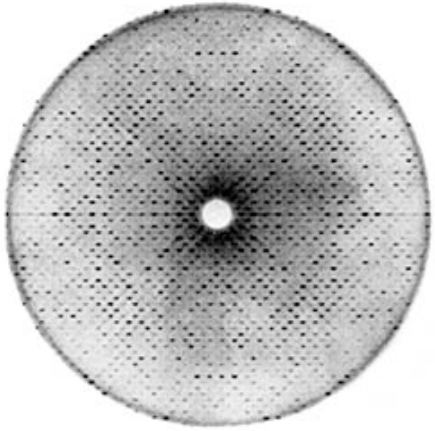
Videos by Lorenzo Labitiga

Imaging is pervasive in structural biology

- The experimental techniques used to determine macromolecular (e.g., protein) structure also depend on imaging

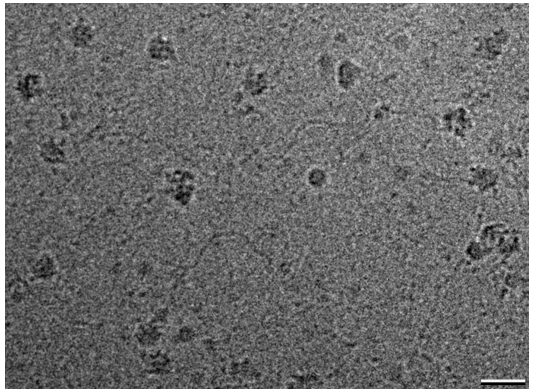
The outputs of these images are less obvious, we cannot simply view the image and understand what the item being imaged looks like. In order to extract information, we have to use image processing and reconstruction tools.

X-ray crystallography



https://askabiologist.asu.edu/sites/default/files/resources/articles/crystal_clear/CuZn_SOD_C2_x-

Single-particle cryo-electron microscopy



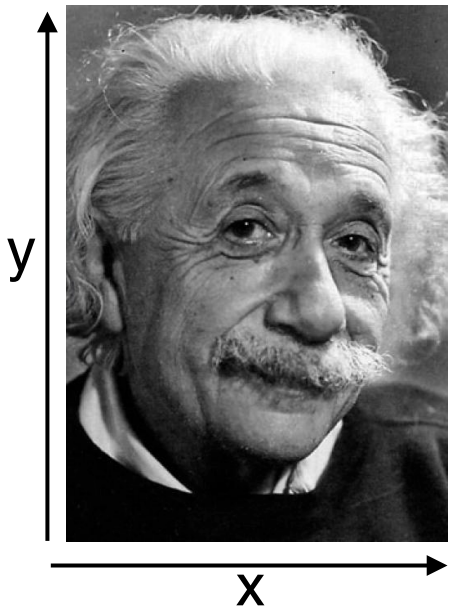
http://debkellylab.org/?page_id=94

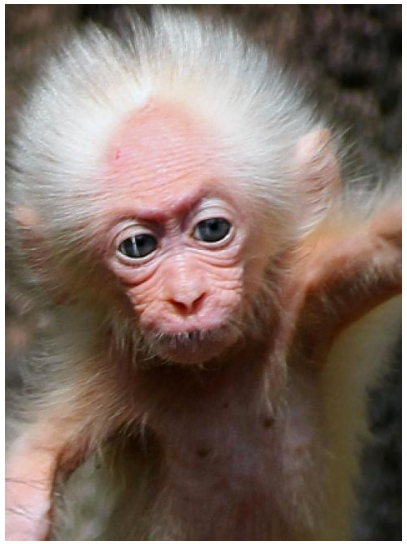
Computation plays an essential role in these imaging-based techniques

- Some techniques require substantial computation before you can even see the image
- We will start with analysis of microscopy data, because it's closest to our everyday experience with images
- In fact, the basic image analysis techniques we'll cover initially also apply to normal photographs

Representations of an image

- Recall that we can think of a grayscale image as:
  - A function of two variables (*x* and *y*)
  - A two-dimensional array of brightness values
  - A matrix (of brightness values)
- A color image can be treated as:
  - Three separate images, one for each color channel (red, green, blue)
  - A function that returns three values (red, green, blue) for each (*x*, *y*) pair



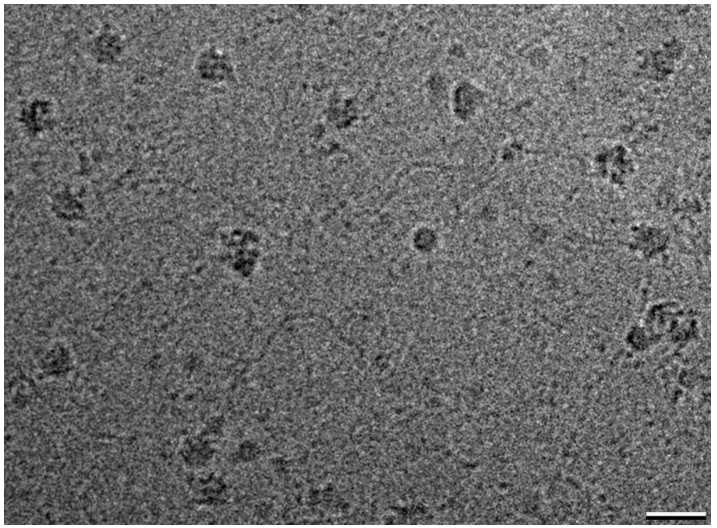


Reducing image noise

Experimentally determined images are always corrupted by *noise*

- "Noise" means any deviation from what the image would ideally look like

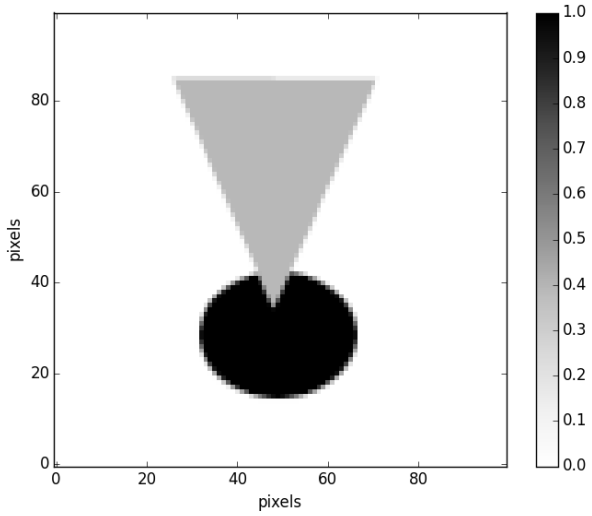
There is always a trade-off between signal and noise, as noise removal algorithms will also remove signal. The end-goal of noise-reduction methods is to remove as much noise as possible, while retaining as much signal as possible. Signal refers to features of an image (borders, shapes, objects, fluorescence, etc) that we are interested in analyzing.



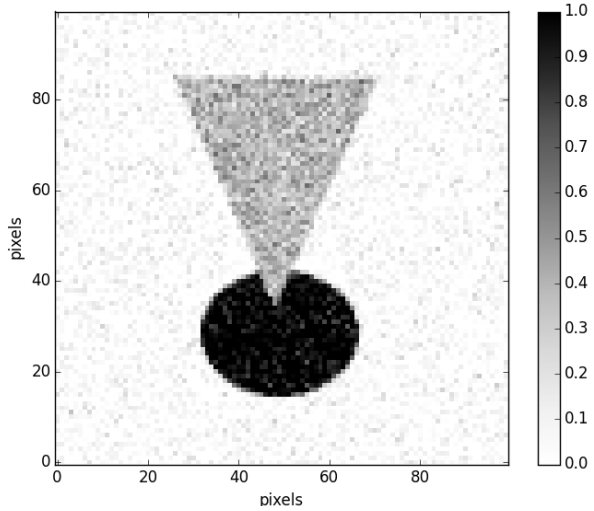
http://www.siox.org/pics/pferd-noisy.jpg

Image noise

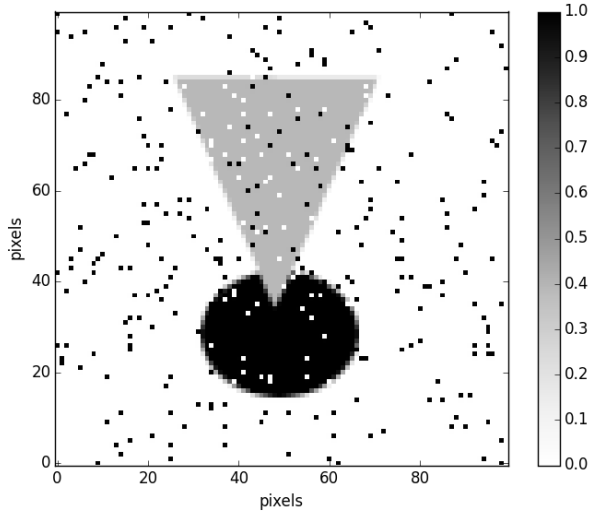
Original image



Noisy images



"Gaussian noise": normally distributed noise added to each pixel



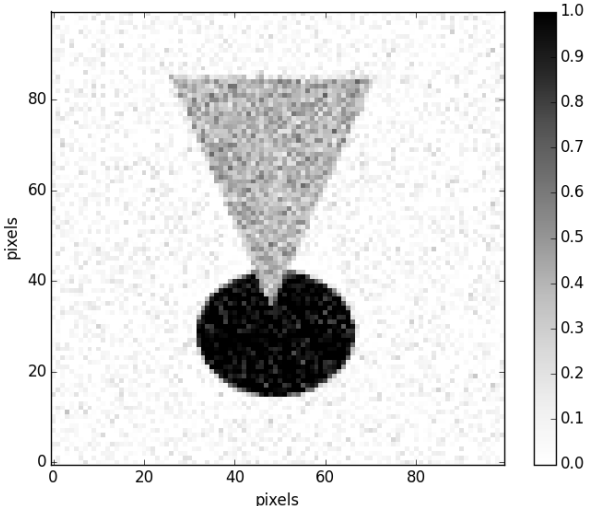
"Salt and peper noise": random pixels replaced by very bright or dark values

Image noise

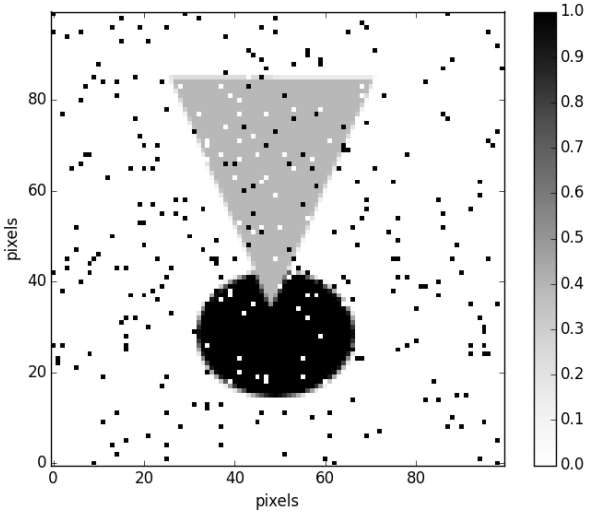
ex. Ambient light noise, aberrations, will lead to gaussian noise . This type of noise tends to be more obvious when the signal itself is relatively weak. We would say the signal-to-noise ratio (SNR) is low.

ex. Pixels are broken in camera or file corruption can lead to salt-and-pepper noise, in which entire pixels values are meaningless.

Noisy images



"Gaussian noise": normally distributed noise added to each pixel



"Salt and peper noise": random pixels replaced by very bright or dark values

Reducing image noise

Mean and Gaussian filters

How can we reduce the noise in an image?

- The simplest way is to use a “mean filter”
  - Replace each pixel by the average of a set of pixels surrounding it
  - For example, a 3x3 mean filter replaces each pixel with the average of a 3x3 square of pixels
  - This is equivalent to convolving the image with a 3x3 matrix:

We could also say that we are convolving our image by a 3x3 matrix

$$\begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix} = 1/9 \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Why 1/9?

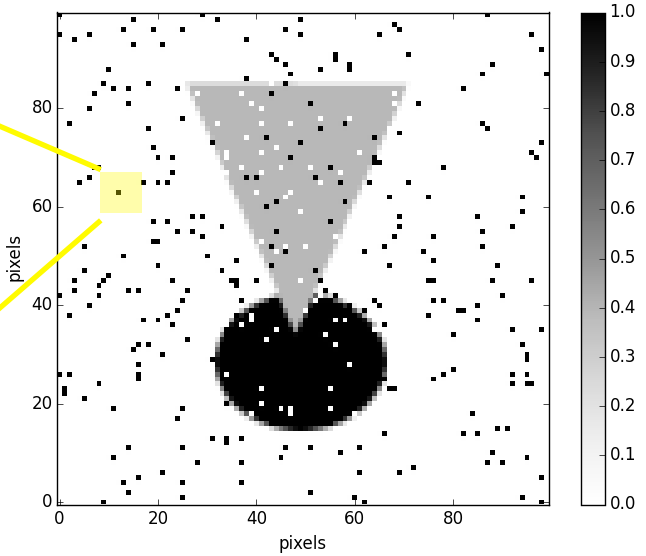
It’s best to use an odd-sized matrix so that there is a center pixel. Then, the pixel you are examining in the original image would be at the exact center of the matrix

Values should sum to 1, so that overall brightness of the image remains constant

Mean filter

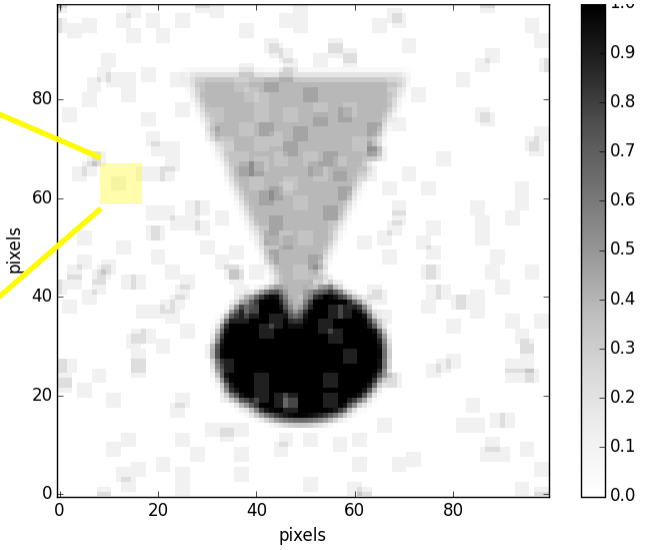
Original images

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



Result of 3x3 mean filter

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & 0 & 0 \\ 0 & 0 & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & 0 & 0 \\ 0 & 0 & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

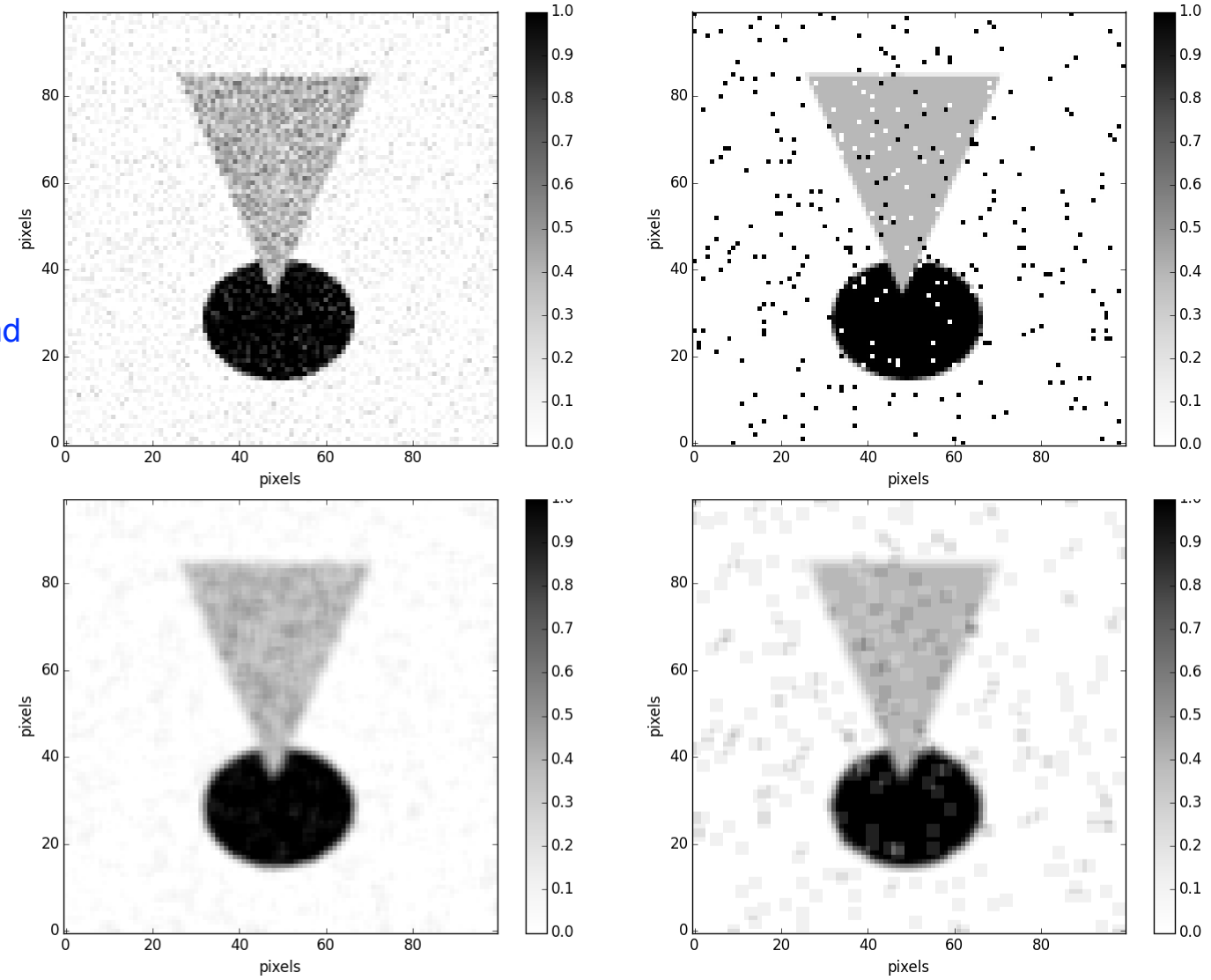


Mean filter

Original images

Trade-off between Noise-reduction and blurring of signal

Result of 3x3 mean filter

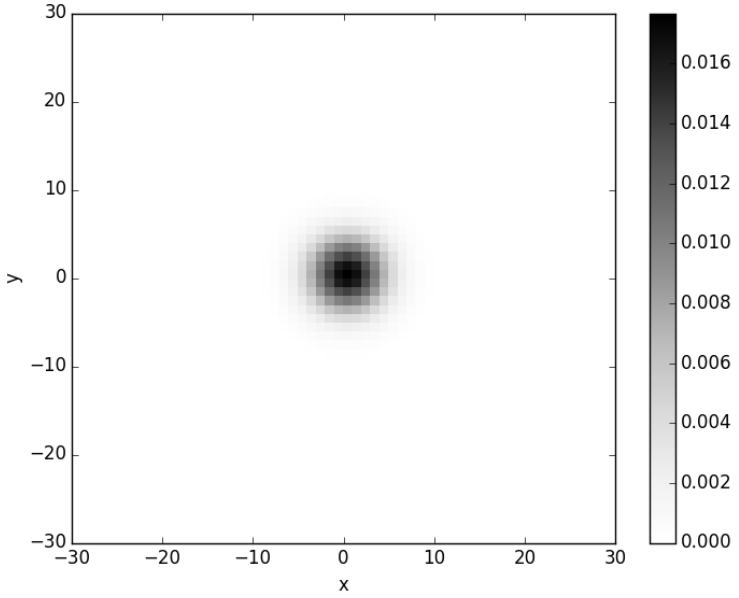


A larger filter (e.g., 5x5, 7x7) would further reduce noise, but would blur the image more

A better choice: use a smoother filter

- We can achieve a better tradeoff between noise reduction and distortion of the noise-free image by convolving the image with a smoother function
- One common choice is a (two-dimensional) Gaussian
- Rather than choosing exact size, choose the standard deviation

Gaussian filter



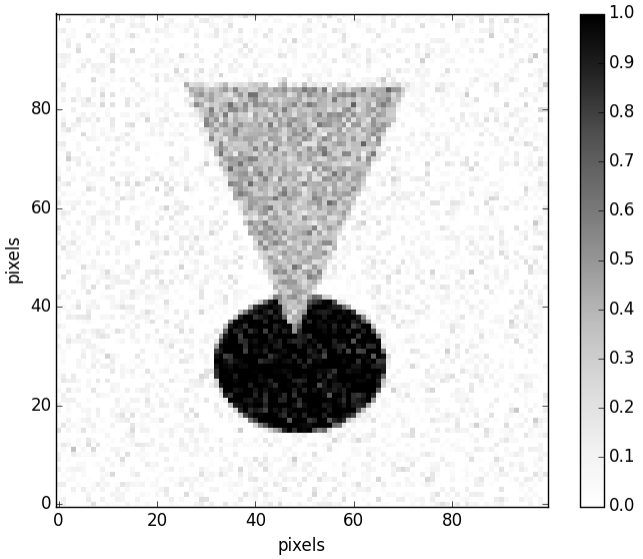
Weights sum to 1, so that the output image has the same as before.

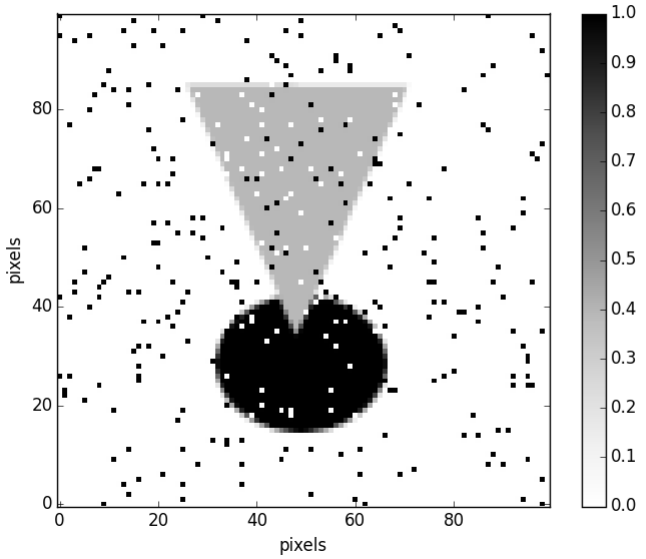
$$\begin{bmatrix} 0.003 & 0.013 & 0.022 & 0.013 & 0.003 \\ 0.013 & 0.059 & 0.097 & 0.059 & 0.013 \\ 0.022 & 0.097 & 0.159 & 0.097 & 0.022 \\ 0.013 & 0.059 & 0.097 & 0.059 & 0.013 \\ 0.003 & 0.013 & 0.022 & 0.013 & 0.003 \end{bmatrix}$$

standard deviation = 1 pixel

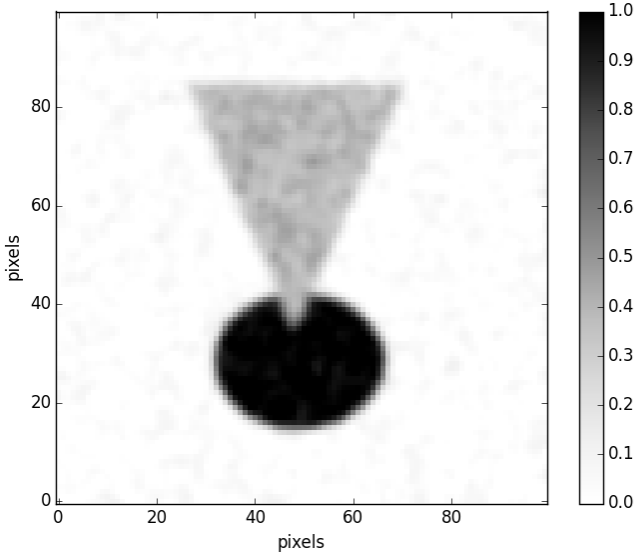
Gaussian filter

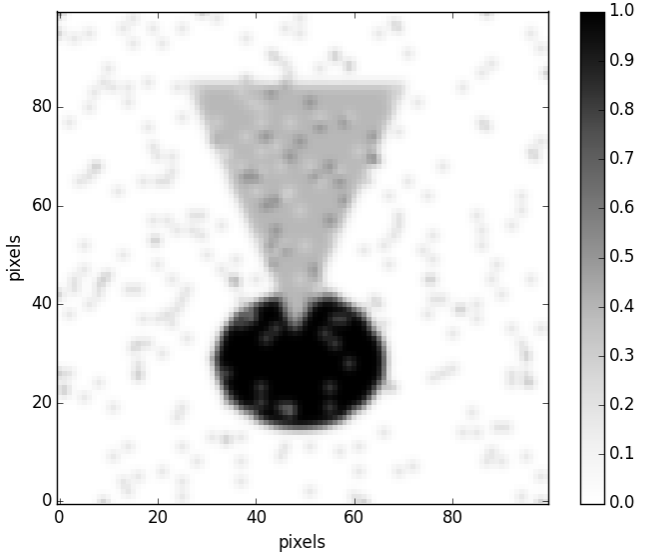
Original images





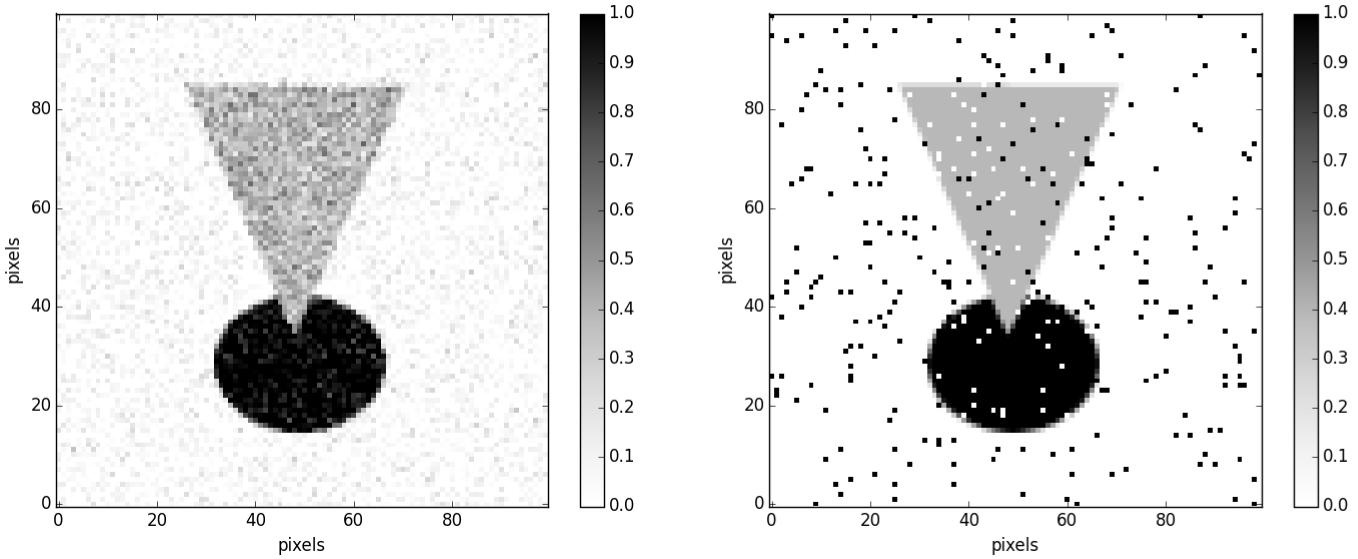
Result of Gaussian filter (standard deviation σ = 1 pixel)



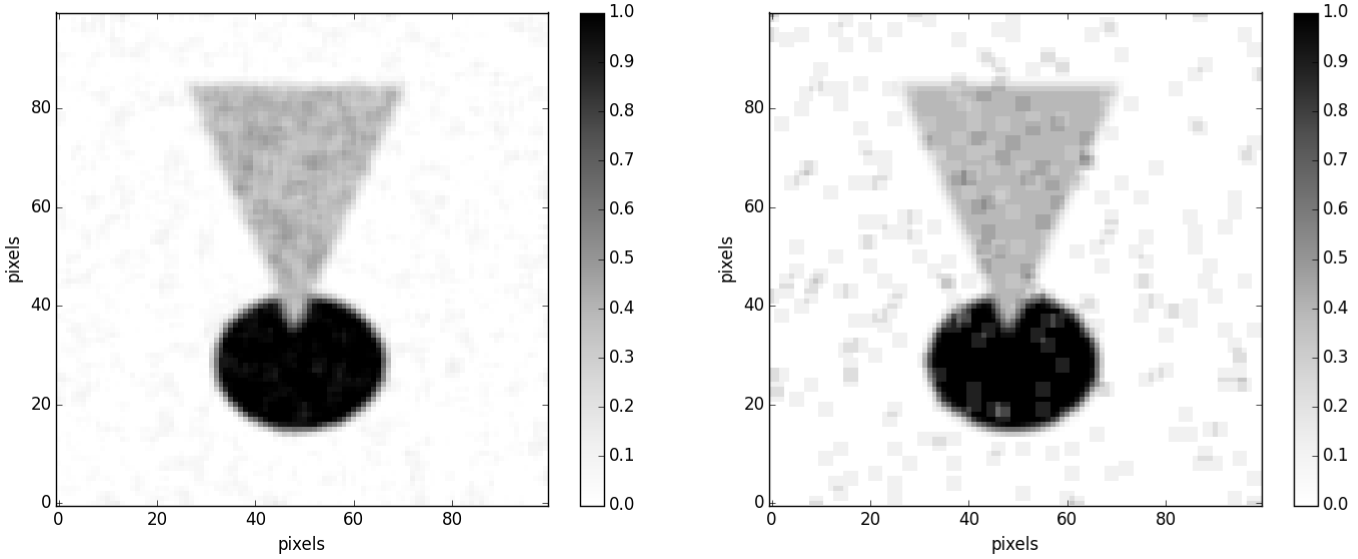


Mean filter (for comparison)

Original images

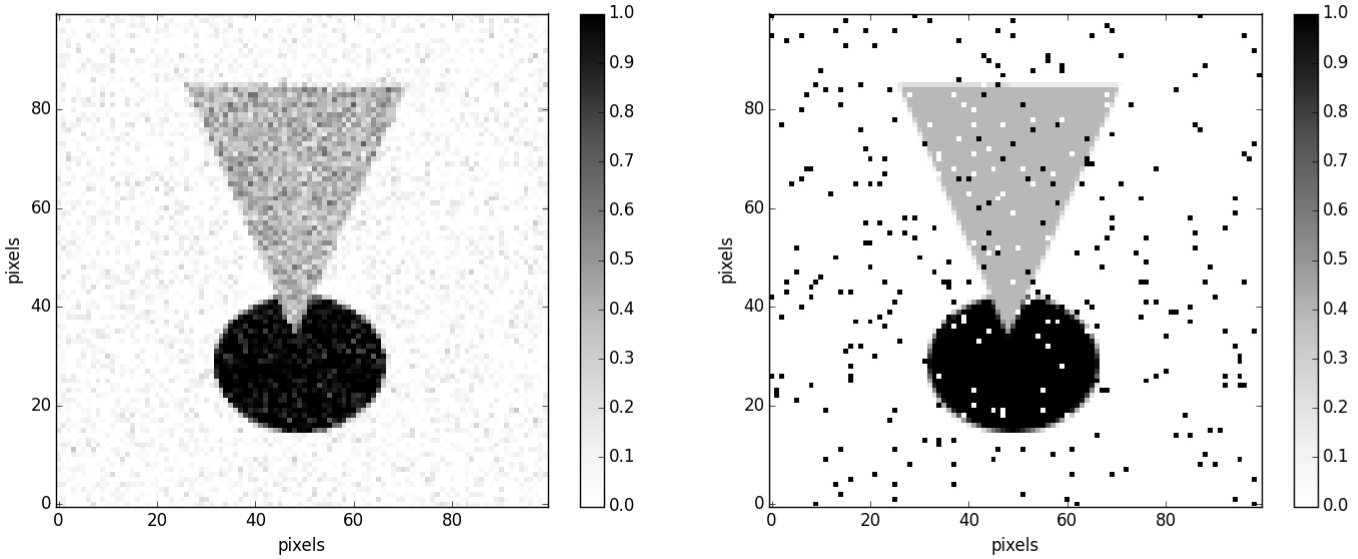


Filtered images

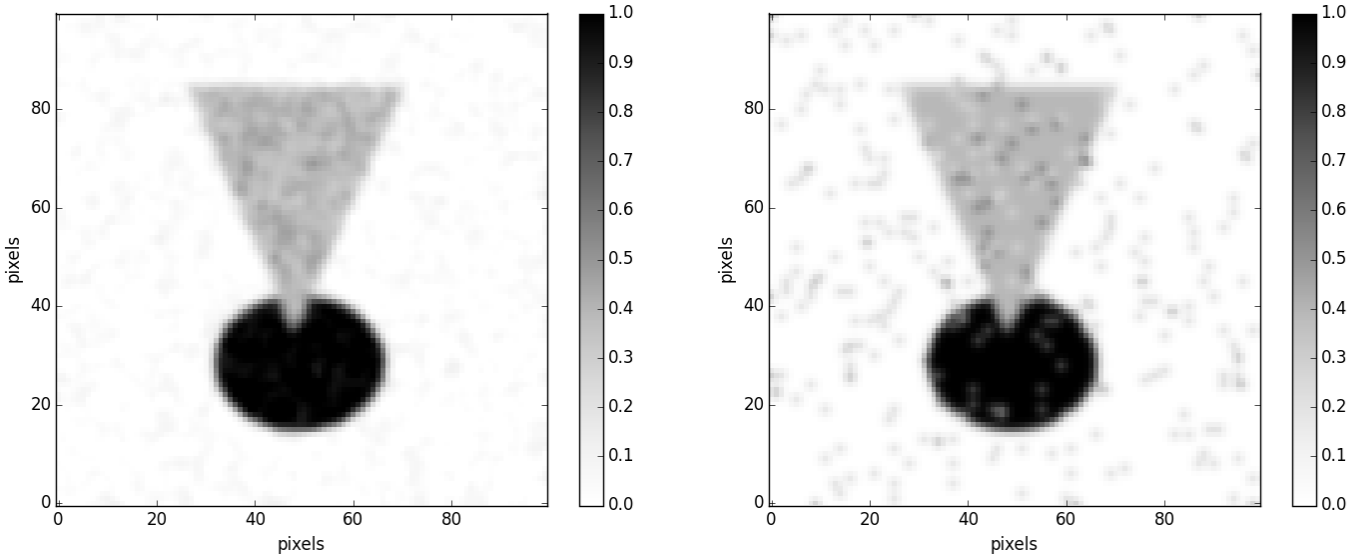


Gaussian filter (for comparison)

Original images



Filtered images



Reducing image noise

Frequency domain interpretation

Low-pass filtering

- Because the mean and Gaussian filters are convolutions, we can express them as multiplications in the frequency domain
- Both types of filters reduce high frequencies while preserving low frequencies. They are thus known as *low-pass filters*
- These filters work because real images have mostly low-frequency content, while noise tends to have a lot of high-frequency content

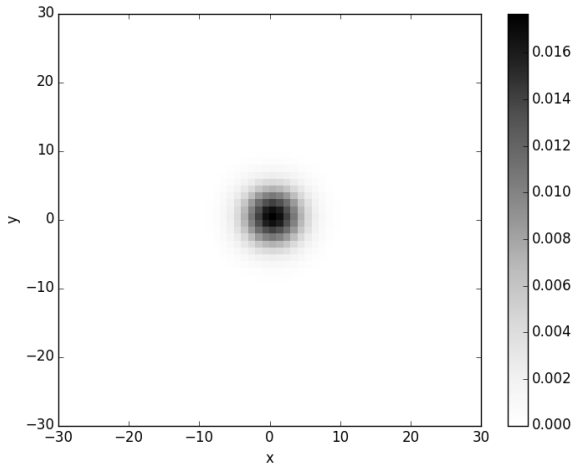
Low-frequencies pass through (are preserved) while higher frequencies do not

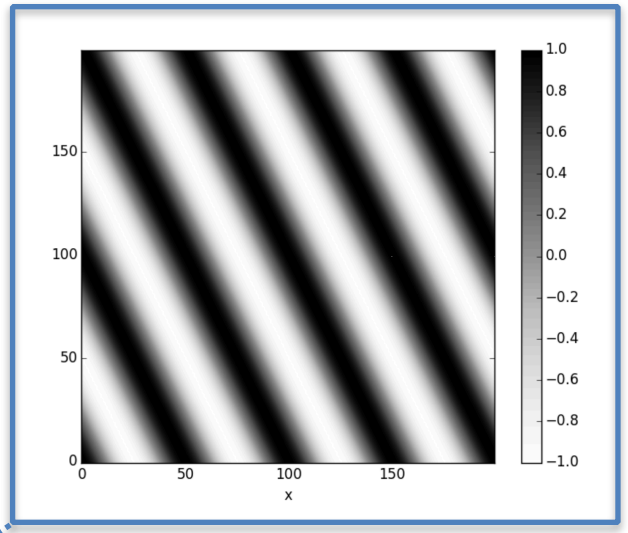
Note that boundaries in images are made up of higher frequencies and that is why blurring of borders/edges occurs when you apply these filters

Low-pass filtering

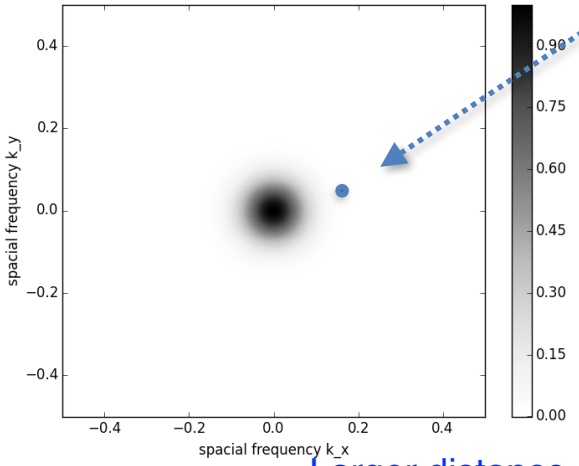
Filter in real domain

Gaussian filter





Magnitude profile in frequency domain (low frequencies are near center of plots)



Point value in the frequency domain describes the strength of the corresponding frequency in the real domain

Larger distance away from origin indicates higher frequencies

Angle of vector from origin represents direction of sine wave, or the relative x and y frequencies

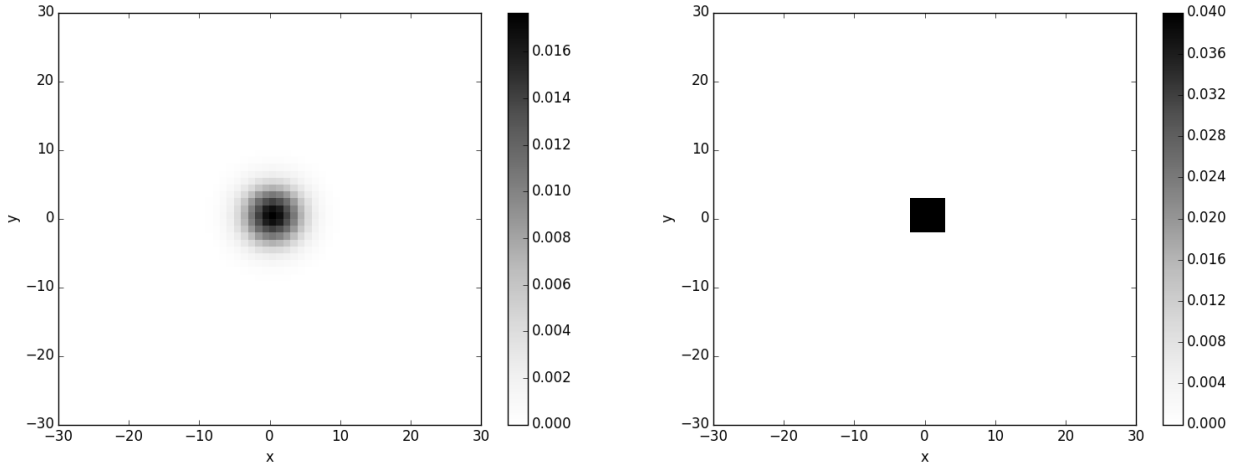
Fourier transform of a gaussian (in the position domain) turns out to also be a gaussian (in the frequency domain)

Low-pass filtering

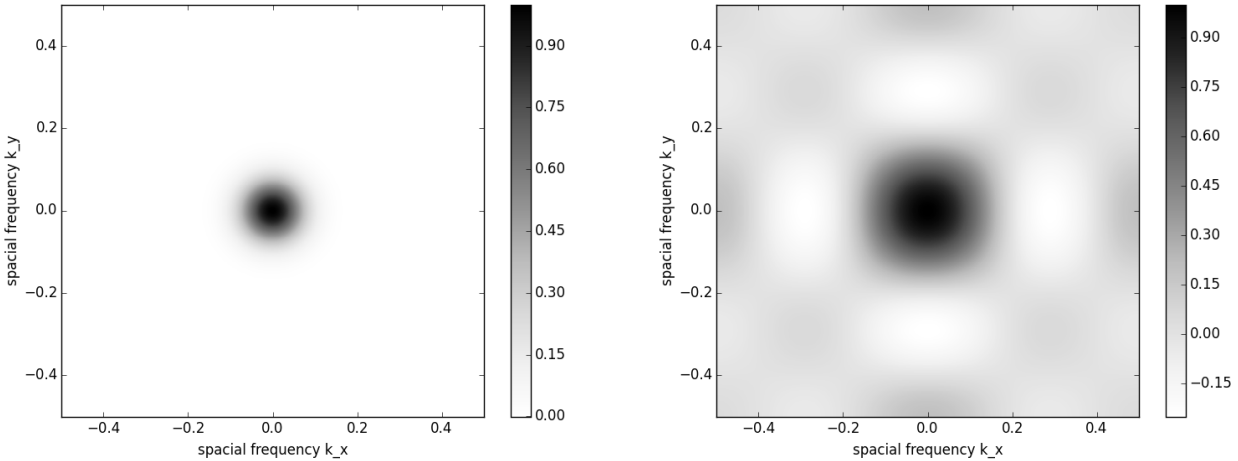
Gaussian filter

Mean filter

Filter in real domain



Magnitude profile in frequency domain
(low frequencies are near center of plots)



The Gaussian filter eliminates high frequencies more effectively than the mean filter, making the Gaussian filter better by most measures.

Low-pass filtering

- As a filter becomes larger (wider), its Fourier-domain representation becomes narrower
- In other words, making a mean or Gaussian filter larger will make it more low-pass (i.e, narrow the range of frequencies it passes)
  - Thus it will eliminate noise better, but blur the original image more

Reducing image noise

Median filter

Median filter

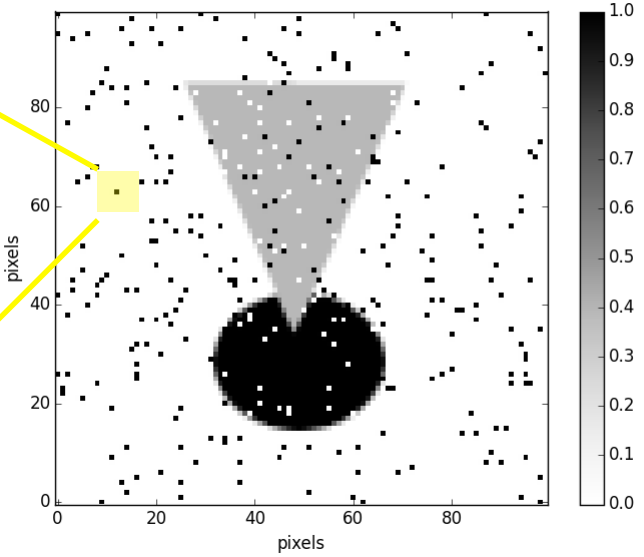
- A median filter ranks the pixels in a square surrounding the pixel of interest, and picks the middle value
- This is particularly effective at eliminating noise that corrupts only a few pixel values (e.g., salt-and-pepper noise)
- This filter is *not* a convolution

  ^ This is because a convolution is applied the exact same way on each region/window of the image, however, the median operations depends on what values are within that region, and thus will not be the same each time

Median filter

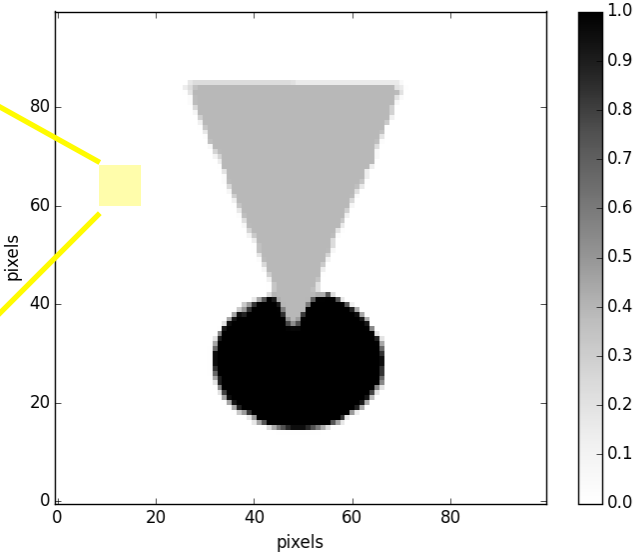
Original images

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



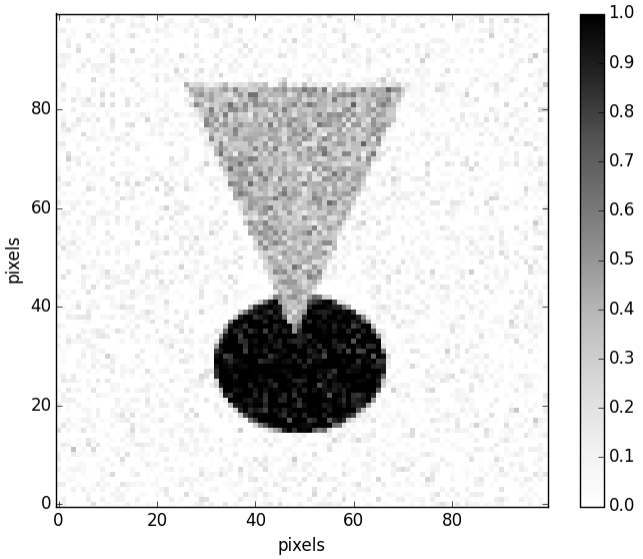
Result of 3x3 median filter

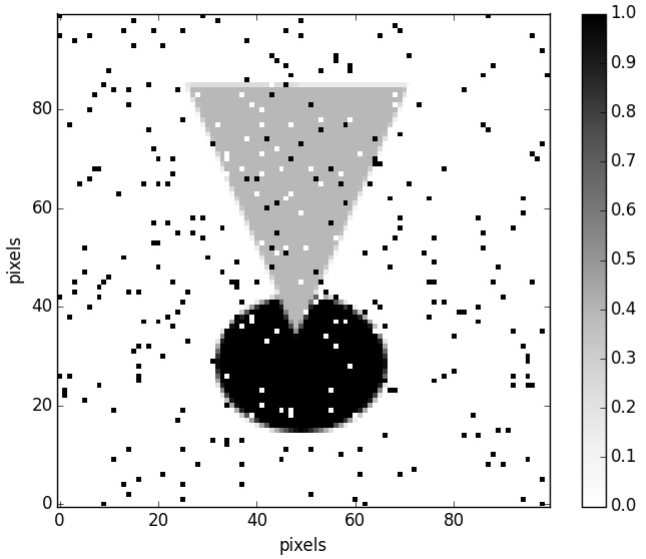
$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



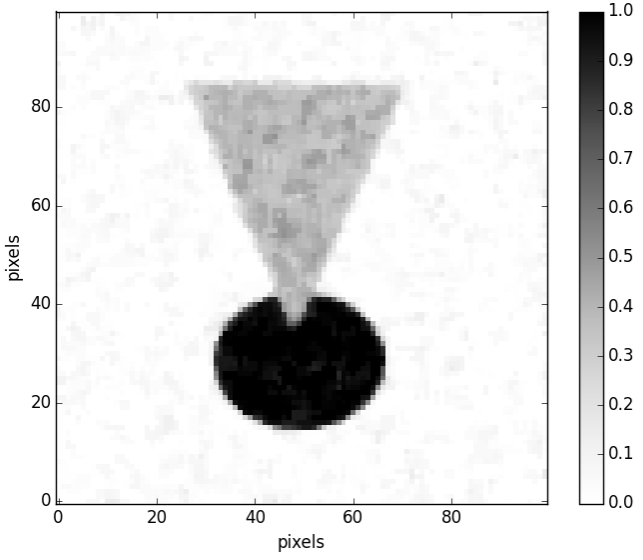
Median filter

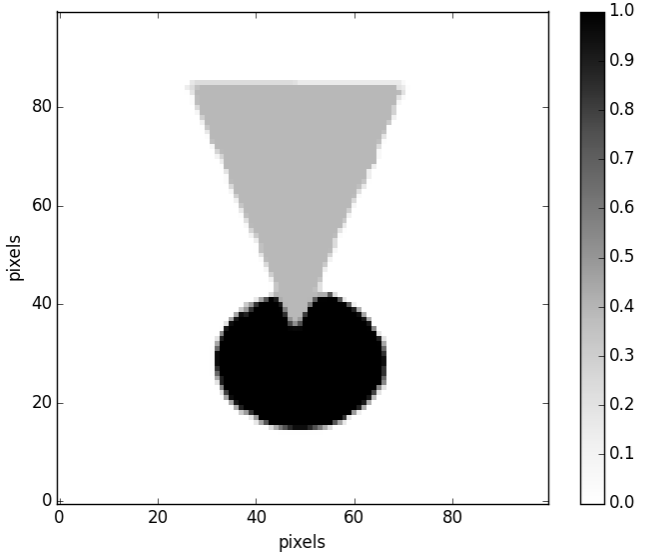
Original images





Result of 3x3 median filter





Using a larger window would further reduce noise, but would distort the image more

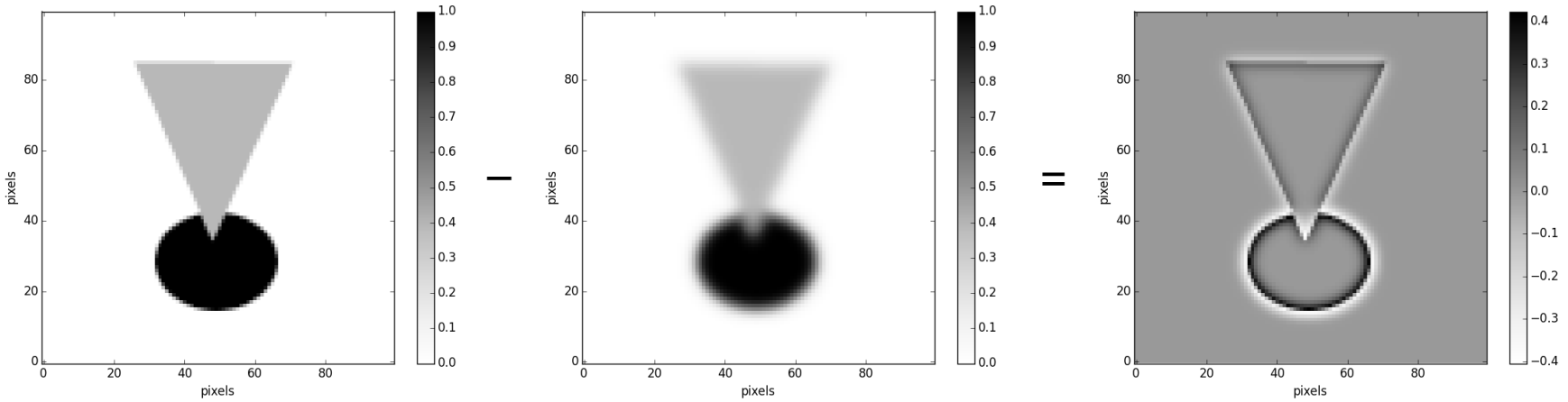
Sharpening images

High-pass filter

- A *high-pass filter* removes (or reduces) low-frequency components of an image, but not high-frequency ones
- The simplest way to create a high-pass filter is to subtract a low-pass filtered image from the original image
  - This removes the low frequencies but preserves the high ones
  - The filter matrix itself can be computed by subtracting a low-pass filter matrix (that sums to 1) from an "identity" filter matrix (all zeros except for a 1 in the central pixel)

High-pass filtering is the same as subtracting the low-pass image from the original

Original image | Low-pass filtered image | High-pass filtered image



High-pass filter

Horizontal edge in original image

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 9 & 9 & 9 & 9 & 9 \\ 9 & 9 & 9 & 9 & 9 \\ 9 & 9 & 9 & 9 & 9 \end{bmatrix}$$

−

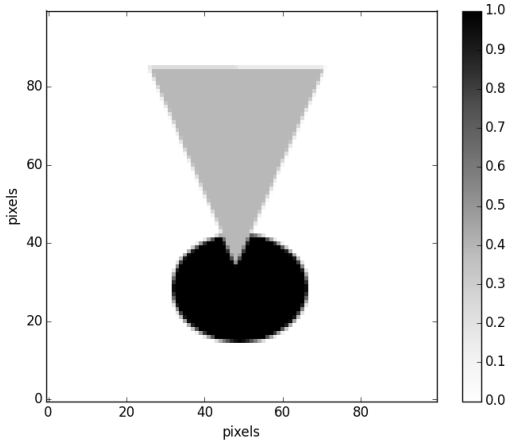
Low-pass filtered (3 x 3 mean filter)

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 3 & 3 & 3 & 3 & 3 \\ 6 & 6 & 6 & 6 & 6 \\ 9 & 9 & 9 & 9 & 9 \\ 9 & 9 & 9 & 9 & 9 \end{bmatrix}$$

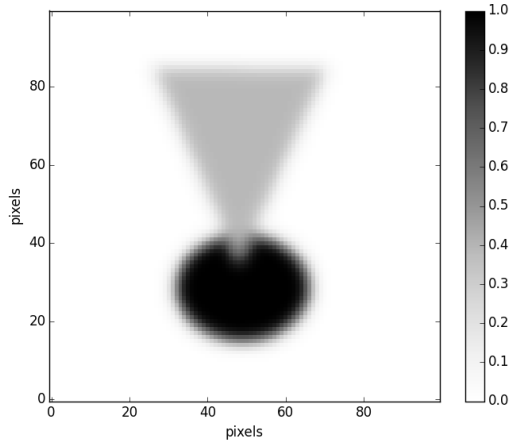
=

High-pass filtered (edges are emphasized)

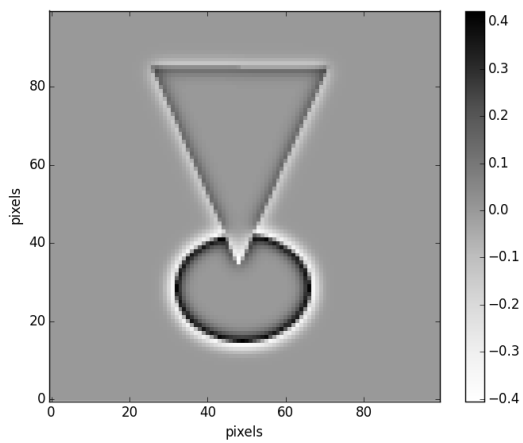
$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -3 & -3 & -3 & -3 & -3 \\ 3 & 3 & 3 & 3 & 3 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



−



=



How might one use a high-pass filter?

- To highlight edges in the image
- To remove any “background” brightness that varies smoothly across the image

^ This kind of effect is low-frequency

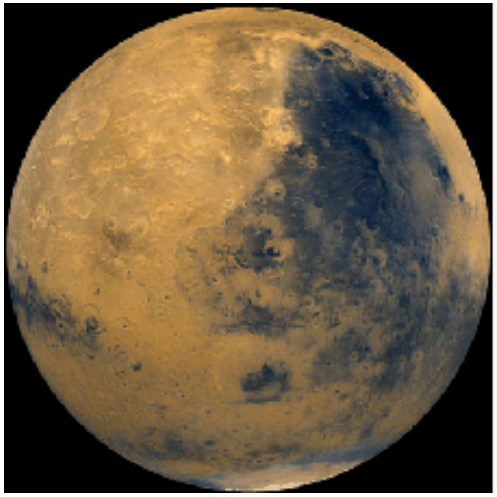
- Image sharpening
  - To sharpen the image, one can add a high-pass filtered version of the image (multiplied by a fractional scaling factor) to the original image
  - This increases the high-frequency content relative to low-frequency content
  - In photography, this is called “unsharp masking”

Original image | Mild sharpening (small scale factor) | Stronger sharpening (larger scale factor)

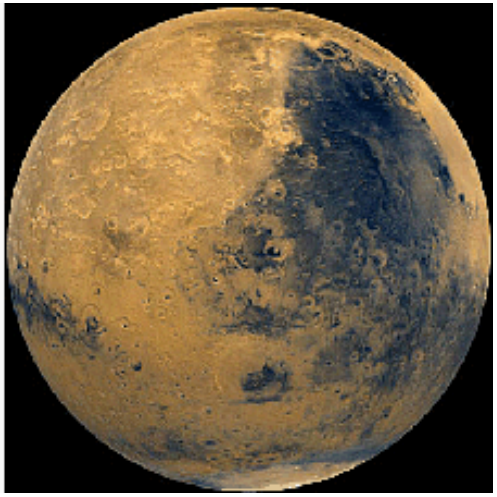
https://upload.wikimedia.org/wikipedia/commons/0/0b/Accutance_example.png

Image sharpening — another example

Original image



Sharpened image



http://www.exelisvis.com/docs/html/images/unsharp_mask_ex1.gif

Image description and classification

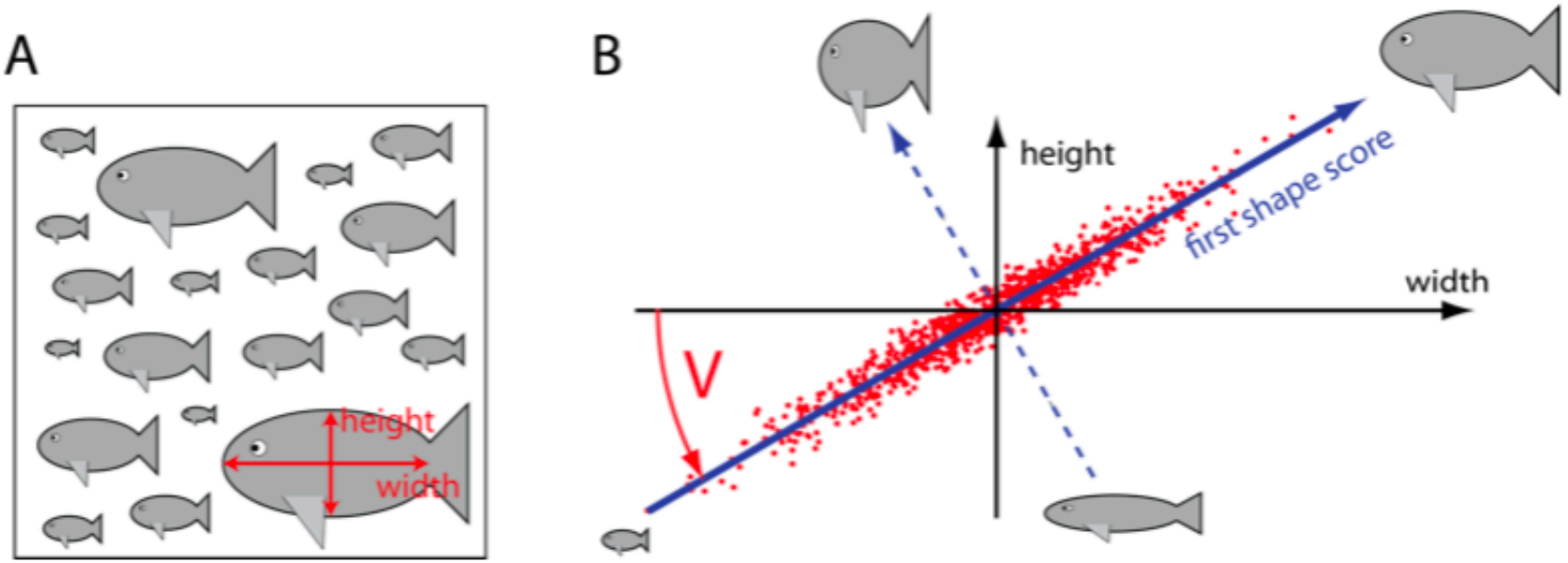
Describing images concisely

- The space of all possible images is very large. To fully describe an *N*-by-*N* pixel grayscale image, we need to specify N^2 pixel values.
- We can thus think of a single image as a point in an N^2-dimensional space.
- Classifying and analyzing images becomes easier if we can describe them (even approximately) with fewer values.
- For many classes of images, we can capture most of the variation from image to image using a small number of values
  - This allows us to think of the images as points in a lower-dimensional space
  - We'll examine one common approach: Principal Components Analysis.

How can we compress the images down to a lower dimensional space?

Principal component analysis (PCA)

- Basic idea: given a set of points in a multi-dimensional space, we wish to find the linear subspace (line, plane, etc.) that best fits those points.

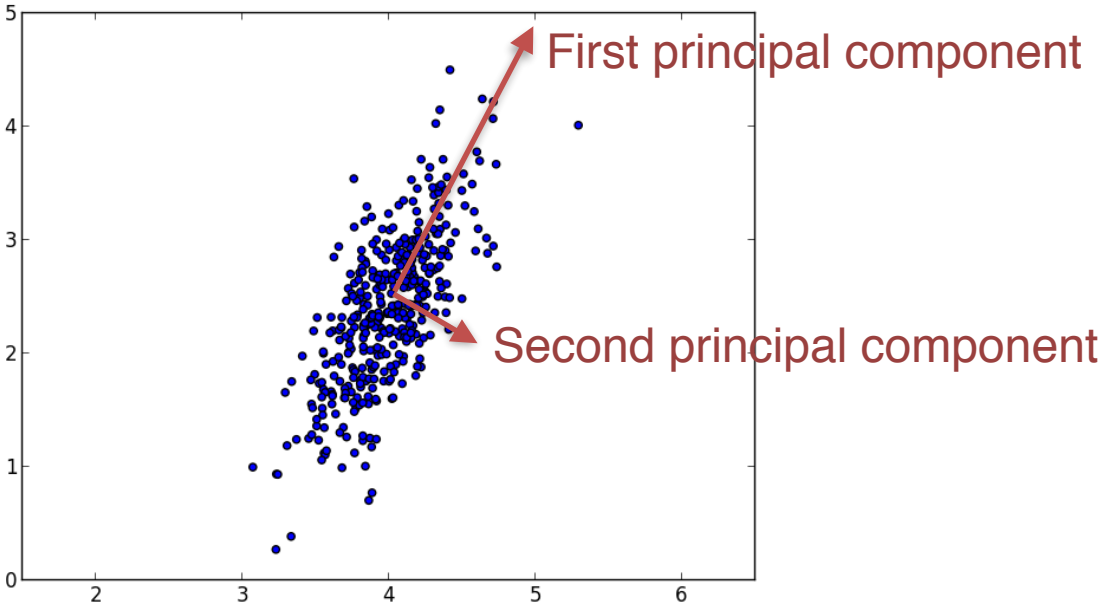


How can we define the parameters that capture the most information in the data?

Werner et al., PLoS ONE 2014

Principal component analysis (PCA)

- Basic idea: given a set of points in a multi-dimensional space, we wish to find the linear subspace (line, plane, etc.) that best fits those points.



- If we want to specify a point ***x*** with just one number (instead of two), we can specify the closest point to ***x*** on the line described by the first principal component (i.e., project ***x*** onto that line).
- In a higher dimensional space, we might specify the point lying closest to ***x*** on a plane specified by the first two principal components.

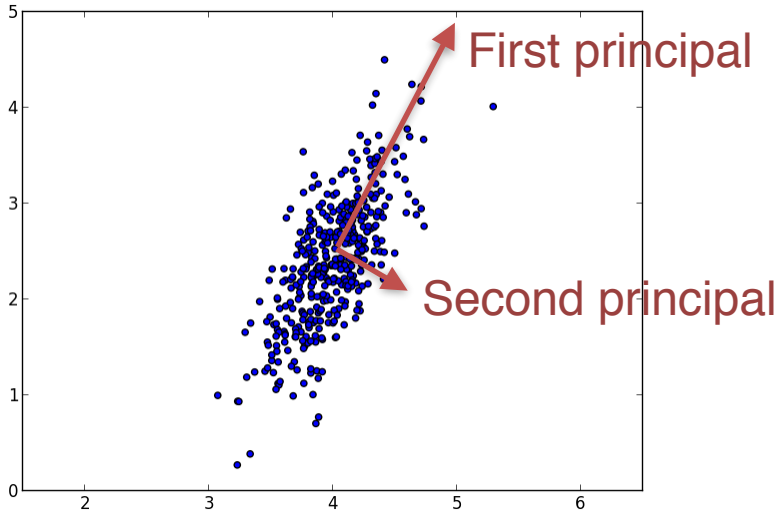
Principal component analysis (PCA)

- How do we pick the principal components?
  - First subtract off the mean value of the points, so that the points are centered around the origin.
  - The first principal component is chosen to minimize the sum squared distances of the points to the line it specifies.
  - This is equivalent to picking the line that maximizes the variance of the full set of points after projection onto that line.
  - The *k*th principal component is calculated the same way, but required to be orthogonal to previous principal components

ie, we want to pick a line, such that there is the most "spread"/greatest range of data points on that line

Enforcing orthogonality means that the variation/information we capture in each component won't overlap

We stop finding additional components once the variation explained by the next component falls below a certain threshold. Ie we've captured most of the variance in the data using the components we've already found



Example: face recognition

- A popular face recognition algorithm relies on PCA
  - Take a large set of face images (centered, frontal views)
  - Calculate the first few principal components
  - Approximate each new face image as a sum of these first few principal components (each multiplied by some coefficient).
  - Classify faces by comparing these coefficients to those of the original face images

First component represents signal from face and neck, but not from background

Original images

Principal components (viewed as images)

http://www.pages.drexel.edu/~sis26/Eigenface%20Tutorial.htm

We are capturing which aspects of the image are most important to recognizing and distinguishing different people

Practical image processing tips

Practical image processing tips

(Thanks to Leo Kesselman)

- When viewing an image, you might need to scale all the intensity values up or down so that it doesn't appear all black or all white
- You might also need to adjust the "gamma correction" to get the image to look right
  - This applies a nonlinear (but monotonically increasing) function to each pixel value
- ImageJ is a great program for looking at scientific images

You're not responsible for this material, but you may find it useful