Lab 7 – image processimg

Useful resources

Matlab image processing toolbox

[x,map]={tiff,gif,bmp}read('filename') read a {tiff,gif,bmp} file (image matrix – storing color indices, and colormap – mapping of integer indices into [R, G, B] vectors)

I=ind2gray(x, map) change an image from a [matrix, colormap] representation into grayscale matrix, suitable for digital processing.

imshow(x,map) or imshow(r,g,b) show an image

colormap(mapname) preset a colormap for showing
(maps: hsv, gray, hot, cool, bone, copper, pink, prism, jet, flag)

colorbar display a colorbar showing value-to-color mapping

freqz2, filter2, fft2, ifft2 2-D analogues for respective 1-D functions

Local matlab additions

imsub show and cut a fragment from an image (hint: do a ind2gray first!)

imfft, imifft 2-D fft pair with zero frequency shifted into center (displays well!)

m=immask(r,type) makes a 'lowpass' or 'highpass' filter mask in the frequency domain
(hint: you must have image fft (X) displayed on your current figure; $0 < r < min(size(X))/2$ is a cutoff radius)

ffted interactive spectrum editor

Running MATLAB

Use IMGLAB.BAT from DOS to start Win with Matlab and local additions.

Experiments

1. display and sketch a 2-D FFT of a 16x16 image consisting of:

   (a) horizontal stripes of 1 pixel width; of 2 pixels width
   (b) vertical stripes of 1 pixel width; of 2 pixels width
   (c) diagonal stripes of 1 pixel width; of 2 pixels width
   (d) checkerboard with 1x1 and 2x2 fields

   hint: `imshow(x); colormap(gray); X=imfft(x); figure; imshow(abs(X)); colormap(jet);`

2. Read a photo image, prepare it for processing (cut a fragment not larger than 150x150); (\CYPS\OBRAZY\GIF\ contains some gifs)
   hint: `[xo,map]=gifread('filename'); imshow(xo); colormap(map); xg=ind2gray(xo,map); x=imsub(xg);imshow(x); colormap(gray); size(x)`

3. Display an FFT of image x (hint: use `abs()` or `log10(abs())`, set colormap to "jet")

4. experiment with linear filters in "time" domain
   `h1=ones(3,3)` and `h2=-ones(3,3); h2(2,2)=8;`
   With each filter:

(a) filter your image
(b) check frequency characteristics (hint: `freqz(h); colormap(jet);`);
(c) describe filter type and visible effects with your own words

5. experiment with linear filters in frequency domain
   Show image FFT and compute maximum mask radius:
   `X=imfft(x); imshow(log10(abs(X))); r=min(size(X))/2;`

   (a) `h1mask=immask(0.3*r,'highpass');imshow(h1mask);`
   (b) `h2mask` - same but lowpass, radius `0.7*r`

   With each filter:

   (a) filter your image by multiplying image FFT with mask; display filtered image
   (b) Write answer: what is the filter order?
   (c) describe filter type and visible effects with your own words

6. experiment with special linear filters

   (a) (choose one) `lapx=ones(3,1)*[-1 2 -1]; lapy=lapx';`
   (b) (choose one) `sobh=[-1 0 1]'*[1 2 1]; sobv=sobh';`

   Try to describe filter characteristics and effects on your image.

optional Removing periodic distortion
Distort your image: `[xst]=imstripe(x)`.
Then try the following and describe efficiency:

   (a) lowpass linear filter of order 3 (`h1` from previous experiments)
   (b) median filter (a nonlinear one: `medfilt2(xst)`; chooses a **median** value from 3x3 area)
   (c) spectrum editor (`ffted`) – remove (zero) spectrum fragments influenced most by the distortion

7. Linear and nonlinear filtering
   Distort your image with:

   (a) "salt and pepper" noise `xsalt=imnoise(x, 'salt and pepper');`
   (b) gaussian noise (zero mean, 0.001 variance) `xgauss=imnoise(x, 'gaussian', 0, 0.001);`

   Try to remove each distortion type with:

   (a) lowpass linear filter of order 3 (`h1` from previous experiments)
   (b) median filter (a nonlinear one: `medfilt2(xst)`; chooses a **median** value from 3x3 area)

   Describe effects and try to analyze them.

8. Space images processing

   (a) read a 3-channel image from earth observing satellite `[r g b]=tiffread('lanier');` Display the image.
   (b) compute histograms for each channel `imhist(r);`. Why is the picture "dim"?
   (c) try to rescale each channel (add constant, multiply by factor) to obtain full 0.0-1.0 scale.
   (d) try to rescale each channel statistically, to cover $\mu - 2\sigma$ till $\mu + 2\sigma$ with display scale of 0.0-1.0.

   optional Use NDVI (normalized digital vegetation index) to display vegetation areas.
   NDVI is defined as $NDVI = \frac{IR-R}{IR+R}$. Infrared channel (IR) is our "red", true red channel (R) is our "green".

File: lab7 LaTeXed on October 12, 2005