

CVIPtools C Functions

This is the most recent listing of all C functions available to CVIPtools developers. The functions are grouped by class and library. There are two classes of libraries within CVIPtools – Toolkit and Toolbox.

The Toolkit libraries contain low-level functions, such as data handling and memory management. The Toolbox libraries contain the functions that are typically used by CVIPlab programmers, such as transforms or segmentation routines. This organization is a hierarchical grouping of libraries in which each class successively builds upon the previous class by using the lower-level functions to create higher-level functions. For more detailed information on a particular function see the associated Help pages in CVIPtools. The source code for the C functions can be found in the *\$CVIPtoolsHOME\CVIPC* directory.

Toolkit Libraries

Band.lib – data handling of multi-spectral imagery

assemble_bands	- assembles multiband image from single band images
bandcast	- cast image data to greater precision
bandcopy	- copy band data
band_minmax	- find the min and max values of each band
extract_band	- extracts one band from a multiband image
matalloc	- allocate an array of matrices
matfree	- free memory allocated by matalloc
vecalloc	- allocate an array of vector
vecfree	- free memory allocated by vecalloc

Image.lib – basic image class methods for type conversion, memory management, etc. (Note: see *\$CVIPtoolsHOME\include\CVIPimage.h* for the *get* and *set* macros, such as *getData_Image*)

cast_Image	- cast an image
delete_Image	- Image class destructor

<code>dump_Image</code>	- print image information
<code>duplicate_Image</code>	- create a new instance of an existing image
<code>getBand_Image</code>	- reference a band of matrix data
<code>getBandVector_Image</code>	- unload image bands into a vector
<code>getColorSpace_Image</code>	- get color space of image (e.g. RGB, GRAY, etc.)
<code>getDataFormat_Image</code>	- get data format (i.e. REAL or COMPLEX)
<code>getDataType_Image</code>	- get data type of image (e.g. CVIP_BYTE, CVIP_FLOAT, etc.)
<code>getData_Image</code>	- returns pointer to data (macro in CVIPimage.h)
<code>getFileFormat_Image</code>	- get file format of image (e.g. PPM, PGM, etc.)
<code>getImagPixel_Image</code>	- read an imaginary pixel sample from the image
<code>getImagRow_Image</code>	- reference an imaginary row of the image
<code>getNoOfBands_Image</code>	- get number of data bands of image
<code>getNoOfCols_Image</code>	- get width of image
<code>getNoOfRows_Image</code>	- get height of image
<code>getPixel_Image</code>	- same as "getRealPixel_Image"
<code>getRealPixel_Image</code>	- read a real pixel sample from the image
<code>getRealRow_Image</code>	- reference a real row of the image
<code>getRow_Image</code>	- same as "getRealRow_Image"
<code>history_add</code>	- add info to image history structure
<code>history_check</code>	- check if an operation has been done on an image
<code>history_copy</code>	- copies information from old_story
<code>history_get</code>	- get info about an operation done on an image
<code>history_show</code>	- setup routine for history print
<code>history_print</code>	- performs output of history structure to h_story
<code>makeComplex_Image</code>	- make real image complex
<code>makeReal_Image</code>	- make complex image real
<code>new_Image</code>	- Image class constructor
<code>setBand_Image</code>	- add a new reference to a band of matrix data
<code>setImagPixel_Image</code>	- write an imaginary pixel sample to the image
<code>setPixel_Image</code>	- same as "setRealPixel_Image"
<code>setRealPixel_Image</code>	- write a real pixel sample to the image data

IO.lib – general purpose Input/Output, memory management routines

<code>allocMatrix3D_CVIP</code>	- allocate memory for volume matrix
<code>allocMatrix_CVIP</code>	- allocate memory for regular matrix
<code>close_CVIP</code>	- close a file for reading or writing
<code>error_CVIP</code>	- print error message to terminal

<code>freeMatrix3D_CVIP</code>	- free memory associated with volume matrix
<code>freeMatrix_CVIP</code>	- free memory associated with regular matrix
<code>getFloat_CVIP</code>	- get floating point value from the user
<code>getInt_CVIP</code>	- get integer value from user
<code>getString_CVIP</code>	- get character string
<code>getUInt_CVIP</code>	- get unsigned integer value from user
<code>init_CVIP</code>	- parse standard info. from command line
<code>msg_CVIP</code>	- print regular message to terminal
<code>openRead_CVIP</code>	- open a file for reading (handles "stdin")
<code>openWrite_CVIP</code>	- open a file for writing (handles "stdout")
<code>perror_CVIP</code>	- print system error message to terminal
<code>print_CVIP</code>	- same as "msg_CVIP" minus extra argument
<code>quiet_CVIP</code>	- turn off messaging
<code>usage_CVIP</code>	- print usage message
<code>verbose_CVIP</code>	- turn on messaging

Mapping.lib – image data mapping functions

<code>condRemap_Image</code>	- if the range is 0-255 no remap is done, if it exceeds this range it is linearly remapped to 0-255
<code>linearTrans_Image</code>	- perform linear mapping of an image through a transformation matrix
<code>logMap_Image</code>	- map image data logarithmically for better display of FFT spectrum
<code>remap_Image</code>	- map image data into a specified range
<code>trun_Image</code>	- remap image data, maintain relative size of each data band

Matrix.lib – matrix algebra, manipulation and numerical analysis routines (Note: see `$CVIPtoolsHOME/include/CVIPmatrix.h` for the *get* and *set* macros, such as *getData_Matrix*)

<code>add_Matrix</code>	- add two matrices
<code>and_Matrix</code>	- perform a bitwise AND on two matrices
<code>cbrt_Matrix</code>	- finds cube root of a matrix (real/complex)
<code>clone_Matrix</code>	- returns a new matrix
<code>conj_Matrix</code>	- find complex conjugate of matrix
<code>copy_Matrix</code>	- copy matrix a to matrix b
<code>covariance_Matrix</code>	- find the covariance estimate of N data bands
<code>crop_Matrix</code>	- create a new matrix from region of original
<code>delete_Matrix</code>	- Matrix class destructor
<code>det_Matrix</code>	- find the determinant of a matrix
<code>duplicate_Matrix</code>	- create new instance of an existing matrix

eigenSystem_Matrix	- find the eigenvectors of a matrix
fastCopy_Matrix	- faster copy if data types are the same
getDataFormat_Matrix	- get data format (i.e. REAL or COMPLEX)
getData_Matrix	- same as "getRealData_Matrix"
getDataType_Matrix	- get data type of matrix (e.g. CVIP_BYTE, CVIP_FLOAT, etc.)
getImagData_Matrix	- reference imaginary data (mapped into rows)
getImagRow_Matrix	- get row of imaginary row
getImagVal_Matrix	- get an "imaginary" matrix element
getNoOfCols_Matrix	- get number of columns in matrix
getNoOfRows_Matrix	- get number of rows in matrix
getRealData_Matrix	- reference real data (mapped into rows)
getRealRow_Matrix	- get row of real data
getRealVal_Matrix	- get a "real" matrix element
getRow_Matrix	- same as "getRealRow_Matrix"
getVal_Matrix	- same as "getRealVal_Matrix"
invert_Matrix	- invert a matrix
mag_Matrix	- find magnitude of a matrix (real/complex)
makeComplex_Matrix	- make real matrix complex
makeReal_Matrix	- make complex matrix real
mult_Matrix	- perform vector multiplication of two matrices
multP Wise_Matrix	- perform piece-wise multiplication
new_Matrix	- Matrix class constructor
print_Matrix	- print contents of matrix in row major form
read_Matrix	- read a matrix structure from disk
rect2pol_Matrix	- convert from rectangular to polar coordinates
scale_Matrix	- scale a matrix by some factor
setImagVal_Matrix	- set an "imaginary" matrix element
setRealVal_Matrix	- set a "real" matrix element
setVal_Matrix	- same as "setRealVal_Matrix"
sqr t_Matrix	- find square root of matrix (real/complex)
square_mag_Matrix	- find magnitude squared of a matrix (real/complex)
sub_Matrix	- subtract two matrices
transpose_Matrix	- find the transpose of a matrix
write_Matrix	- write a matrix structure to disk

Object.lib – object analysis and identification routines

build_ChainCode	- find the chain-code "contour" of an object
-----------------	--

delete_ChainCode	- delete an instance of a chain code object
delete_Object	- object class destructor
drawBB_Objects	- draw a bounding box around all objects
draw_ChainCode	- draw the contour of an object onto an image using the object's chain code
getProp_Object	- find object moment properties
getProp_Objects	- find moment properties of multiple objects
getXY_ChainCode	- turn a chain code into a list of X-Y coordinates
label_Objects	- sequentially label objects (used by label function in libfeature)
listToVector_Objects	- create an object vector from an object list
match_Object	- match an object
new_ChainCode	- create a new instance of a chain code object
new_Object	- object class constructor
print_ChainCode	- print the chain code results to a file
printLabel_Objects	- print an object list to a file
print_Object	- print object statistics to file
printProp_Objects	- print a list of object properties to a file
read_ChainCode	- read a chain code from a file
readLabel_Objects	- read an object list from a file
read_Object	- read object statistics from disk
readProp_Objects	- read a list of object properties from a file
report_ChainCode	- print out the chain code values
trimList_Objects	- trim an object list based on properties

ObjectManager.lib – object managers/handlers

addhead_DLL	- add link to head of list
addnext_DLL	- add link following the current link
addtail_DLL	- add link to tail of list
delete_DLL	- double Linked List class destructor
find_DLL	- find a particular object in the list
head_DLL	- set current link to head of list
isempty_DLL	- is the list empty?
ishead_DLL	- is current link pointing to head?
istail_DLL	- is current link pointing to tail?
new_DLL	- double Linked List class constructor
next_DLL	- point to next link
previous_DLL	- point to previous link

print_DLL	- print list
print_reverse_DLL	- print list in reverse order
promote_DLL	- promote current link to head of list
removecurr_DLL	- remove current link
removehead_DLL	- remove link from head of list
removetail_DLL	- remove link from tail of list
replace_DLL	- replace object pointed to be current link
retrieve_DLL	- retrieve object pointed to by current link
size_DLL	- get size of list (number of links)
tail_DLL	- set current link to tail of list
addhead_LL	- add link to head of list
addnext_LL	- add link following the current link
delete_LL	- linked list class destructor
find_LL	- find a particular object in the list
head_LL	- set current link to head of list
isempty_LL	- is the list empty?
ishead_LL	- is current link pointing to head?
istail_LL	- is current link pointing to tail?
new_LL	- linked list class constructor
next_LL	- point to next link
previous_LL	- point to previous link
print_LL	- print list
promote_LL	- promote current link to head of list
removehead_LL	- remove head link
removenext_LL	- remove next link
replace_LL	- replace object pointed to be current link
retrieve_LL	- retrieve object pointed to by current link
size_LL	- return size of list
tail_LL	- set current link to tail of list
addobject_HT	- add object using separate chaining technique
delete_HT	- hash table class destructor
findobject_HT	- find object
new_HT	- hash table class constructor
setkey_HT	- set the hash table key
isempty_Stack	- determine whether a stack is empty
new_Stack	- create a new instance of an object stack
pop_Stack	- pop on object off of the stack

push_Stack - push an object onto the stack

ROI.lib – region/area of interest designation, manipulation of an image

asgnFullImage_ROI - assign ROI as full image dimension
asgnImage_ROI - assign a ROI to an image
delete_ROI - ROI class destructor
getDataFormat_ROI - get data format of ROI
getDataType_ROI - get data type of ROI
getHorOffset_ROI - get horizontal offset from pixel (0,0)
getHorSize_ROI - get height/horizontal size of region
getImagPixel_ROI - get/read imaginary pixel sample from ROI
getImagRow_ROI - reference imaginary row from the ROI
getNoOfBands_ROI - get number of data bands in ROI
getNoOfCols_ROI - same as "getHorSize_ROI"
getNoOfRows_ROI - same as "getVerSize_ROI"
getPixel_ROI - same as "getRealPixel_ROI"
getRealPixel_ROI - get/read real pixel sample from ROI
getRealRow_ROI - reference real row from the ROI
getRow_ROI - same as "getRealRow_ROI"
getVerOffset_ROI - get vertical offset from pixel (0,0)
getVerSize_ROI - get width/vertical size of region
loadRow_ROI - load data from a buffer into ROI
new_ROI - ROI class constructor
setImagPixel_ROI - set/write imaginary pixel sample to ROI
setPixel_ROI - same as "setRealPixel_ROI"
setRealPixel_ROI - set/write real pixel sample to ROI
unloadRow_ROI - unload row of data from ROI into buffer

Vector.lib – vector algebra and manipulation routines

band2pixel_Vector - convert a band vector to a pixel vector
convolve_Vector - convolve two vectors
copy_Vector - copy vector a to vector b
findHisto_Vector - find the histogram of a vector
findMaxVal_Vector - return maximum value in vector
findMinVal_Vector - return minimum value in vector
normalize_Vector - normalize a vector between 0 and 1
pixel2band_Vector - convert a pixel vector to a band vector

<code>printHisto_Vector</code>	- print histogram values out to a file
<code>subSample_Vector</code>	- sub-sample a list of vector points

Toolbox Libraries

Note: these are for the C functions, for the Matlab CVIP Toolbox M-file functions see associated documents and Help pages

ArithLogic.lib – arithmetic and logical operations on images

add_Image	- add two images
and_Image	- perform a logical AND on two images
divide_Image	- divide one image by another
multiply_Image	- multiply two images
not_Image	- perform a logical NOT on an image
or_Image	- perform a logical OR on two images
subtract_Image	- subtract one image from another
xor_Image	- perform a logical XOR on two images

Color.lib – color map utilities and color transforms

colorxform	- performs seven color transforms, and inverse transforms
ipct	- performs the inverse principal components transform
luminance_Image	- performs color to luminance transform
lum_average	- performs color to monochrome using average of all bands
pct	- performs the principal components transform in RGB-space
pct_color	- handles both forward and inverse PCT
pseudocol_freq	- pseudocolor transform using FFT spectrum and filters

Compression.lib – image compression/data reduction routines

bit_compress	- decomposes grey level image into eight bit planes. Each bitplane is then run-length encoded and stored in a binary file
bit_decompress	- decompresses each binary file (corresponding to a particular bit plane) into corresponding binary images
bit_planeadd	- decompresses bitplane files and add any combinations of bit planes to produce the resultant gray level image.
btc_compress	- compress the image in 4*4 blocks, store it in a binary file
btc_decompress	- decompress the image from the encoded binary file
btc2_decompress	- decompress multilevel BTC encoded image
btc3_decompress	- decompress predictive BTC encoded image
btc2_compress	- multilevel block truncation coding (BTC) image compression

btc3_compress	- predictive BTC compression
dpc_compress	- differential predictive coding compression
dpc_decompress	- differential predictive coding decompression
frac_compress	- fractal compression
frac_decompress	- fractal decompression
glr_compress	- performs gray level run length coding for any window length specified by the user (window range 1-125)
glr_decompress	- perform gray level run length decoding from the encoded binary file
huf_compress	- perform Huffman coding and store the probability table and encoded data into a binary file
huf_decompress	- perform Huffman decoding from the encoded binary file
jpg_compress	- JPEG compression
jpg_decompress	- JPEG decompression
jp2_compress	- JPEG2000 compression
jp2_decompress	- JPEG2000 decompression
rms_error	- calculates root-mean-square error between two images
snr	- calculates peak signal-to-noise ratio in decibels
vq_compress	- vector quantization compression
vq_decompress	- vector quantization decompression
xvq_compress	- transform/vector_quantization compression
xvq_decompress	- transform/vector_quantization decompression
zon_compress	- zonal coding based compression, DC quantize
zon_decompress	- zonal coding based decompression, DC quantize
zon2_compress	- zonal coding based compression, no DC quantize
zon2_decompress	- zonal coding based decompression, no DC quantize
zvl_compress	- Ziv-Lempel compression
zvl_decompress	- Ziv-Lempel decompression

Conversion.lib – image conversion, I/O utilities

bintocvip	- convert binary (raw) file format to CVIPtools data structure
bmptocvip	- convert BMP (Windows raw data) to CVIPtools data structure
ccctocvip	- convert CCC file format to CVIPtools data structure
cviptoiris	- convert SGI IRIX file format to CVIPtools data structure
CVIPhalftone	- quantizes gray image to binary, dithering options
cviptobin	- convert CVIPtools data structure to binary (raw) file format
cviptobmp	- convert CVIPtools data structure to BMP file format
cviptoccc	- convert CVIPtools data structure to CCC file format

cviptoeeps	- convert CVIPtools data structure to EPS file format
cviptoitek	- convert CVIPtools data structure to ITEX file format
cviptojpg	- convert CVIPtools data structure to JPEG file format
cviptojpg2	- convert CVIPtools data structure to JPEG2000 file format
cviptopnm	- convert CVIPtools data structure to PNM file format
cviptopng	- convert CVIPtools data structure to PNG file format
cviptoras	- convert CVIPtools data structure to Sun RAS file format
cviptotiff	- convert CVIPtools data structure to TIFF file format
cviptovip	- convert CVIPtools data structure to VIP file format
epstocvip	- convert EPS file format to CVIPtools data structure
giftocvip	- convert GIF file format to CVIPtools data structure
gray_binary	- converts natural binary code to gray code and gray to binary
iristocvip	- convert SGI IRIX file format to CVIPtools data structure
itextocvip	- convert ITEX file format to CVIPtools data structure
jpgtocvip	- convert JPEG file format to CVIPtools data structure
jp2tocvip	- convert JPEG2000 file format to CVIPtools data structure
pnmtocvip	- convert PNM file format to CVIPtools data structure
pngtocvip	- convert PNG file format to CVIPtools data structure
rastocvip	- convert Sun RAS file format to CVIPtools data structure
read_Image	- read image from disk
tifftocvip	- convert TIFF file format to CVIPtools data structure
viptocvip	- convert VIP file format to CVIPtools data structure
write_Image	- write image to disk

Display.lib – display and view functions

view_Image	- general purpose image view function for CVIPlab (note: it is a C/C++ function)
------------	--

Feature.lib - feature extraction and pattern classification functions

area	- find area of binary object (number of pixels)
aspect	- find aspect ratio (based on bounding box) of binary object
centroid	- find row and column coordinates of a binary object
euler	- find Euler number of a binary object
hist_feature	- find histogram features: mean, standard deviation, skew, energy, entropy
irregular	- find irregularity (1/thinness ratio) of binary object
k_nearest_neighbor	- finds the kth nearest neighbor using a distance or similarity measure

label	- labels connected objects in an image
laws_texture	- finds Laws texture features
maximum_value_metric	- calculates the maximum value distance metric between two vectors
min_max_normalization	- normalizes a 2D matrix using min-max normalization method
minkowski_distance	- calculates the Minkowski distance between two (feature) vectors
nearest_centroid	- nearest centroid classification using training and test set
nearest_centroid_set	- calculates centroid for classes in training set
nearest_neighbor	- nearest neighbor classification using training and test set
orientation	- finds orientation of a binary object via axis of least second moment
pattern_city_block	- calculates city block distance between two (feature) vectors
pattern_euclidean	- calculates Euclidean distance between two (feature) vectors
pattern_maximum	- calculates max distance between two (feature) vectors
pattern_min_max_normalization	- uses the min-max normalization method to normalize the training and test files
pattern_minkowski	- calculates city block distance between two (feature) vectors
pattern_range_normalize	- performs range normalization on as set of (feature) vectors
pattern_softmax_scaling_normalization	- uses the softmax scaling method to normalize the training and test files
pattern_standard_normal_density_normalization	- uses the standard normal density method to normalize the training and test set files
pattern_tanimoto	- calculates Tanimoto similarity metric for training and test sets
pattern_unit_vector_normalization	- uses the unit vector normalization method to normalize the training and test set files
pattern_vector_inner_product	- uses the vector inner product normalization method to normalize the training and test set files
perimeter	- find the perimeter length of a binary object
projection	- find row and column projections of size normalized object
range_normalize	- range normalizes a 2-D matrix
rst_invariant	- finds seven rotation/scale/translation-invariant moment based on features of binary object
softmax_scaling	- normalizes a 2D matrix using softmax scaling normalization method
spectral_feature	- finds spectral features based on FFT power in rings and sectors
standard_normal_density_normalization	- normalizes a 2D matrix using standard normal density normalization method
tanimoto_metric	- calculates Tanimoto metric for two vectors
texture	- finds up to 14 texture features for four orientations
texture2	- new texture function that finds five texture features for four

	orientations
thinness	- finds thinness ratio of binary object
unit_vector_normalization	- normalizes a 2D matrix using unit vector normalization method
vector_inner_product	- calculates normalized vector inner product between two vectors

Geometry.lib – geometry manipulation routines

bilinear_interp	- shrinks or enlarges an image using bilinear interpolation to calculate the gray-level value of new pixels
copy_paste	- copy a subimage from one image and paste it into another
create_black	- create an all-black image
create_cosine	- create a cosine wave image of any size and desired frequency
create_checkboard	- create a checkerboard image
create_circle	- create a binary image of a circle
create_ellipse	- create a binary image of an ellipse
create_line	- create a line image
create_rectangle	- create a rectangular image
create_sine	- create a sine wave image of any size and desired frequency
create_squarewave	- create a square wave image of any size and desired frequency
crop	- crop a subimage from an image
display_mesh	- displays a mesh file as an image, used in warp
enlarge	- enlarges image to a user-specified size
keyboard_to_mesh	- creates a mesh structure from keyboard entry for mesh_warping
mesh_warping	- geometrically distort an image
mesh_to_file	- saves a mesh structure to a file, used in warp
object_crop	- special crop function used in <i>border mask</i> and <i>border image</i>
rotate	- rotate the given image by an angle specified by the user (Range 1 ~ 360 degrees)
shrink	- shrinks the given image by a factor specified by the user (Range 0.1 ~ 1)
solve_c	- solves bilinear equation, used with warp
spatial_quant	- quantize an image by one of the following methods: average, median, or decimation
translate	- move the entire image horizontally and/or vertically; also used for cut-and-paste of subimage
zoom	- zoom the given image by a factor specified by the user (Range 1 ~ 10)

Histogram.lib – image histogram modification/contrast manipulation routines

define_histogram	- allows user to specify an equation for histogram modification
get_histogram	- generates a histogram array from an image
get_histogram_Image	- generates a histogram Image from an image
gray_linear	- gray level linear modification
gray_multiply	- remap (if necessary) to byte and multiply, clip at 255
gray_multiply2	- cast image to float, multiplies by constant, outputs float image
histeq	- histogram equalization
histogram_show	- prints ASCII representation of a histogram
hist_spec	- perform histogram manipulation using formula specified in by character string(s) for the equation(s)
histogram_spec	- perform histogram manipulation using formula specified by define_histogram
hist_slide	- histogram slide
local_histeq	- local histogram equalization
hist_stretch	- histogram stretch, specify range and percent clip both ends
make_histogram	- generates an image of a histogram
showMax_histogram	- creates an image of a histogram of an image

Morphological.lib – morphological image processing routines

MorphClose_Image	- perform grayscale morphological closing
MorphClose	- perform grayscale morphological closing
MorphDilate_Image	- perform grayscale morphological dilation
MorphDilate	- perform grayscale morphological dilation
MorphErode_Image	- perform grayscale morphological erosion
MorphErode	- perform grayscale morphological erosion
morph_hitmiss	- perform morphological hit-or-miss transform
morphIterMod_Image	- perform iterative morphological modification of an image
morpho	- performs iterative modification of an image
MorphOpen_Image	- perform grayscale morphological opening
MorphOpen	- perform grayscale morphological opening
morph_skeleton	- perform morphological skeletonization
morph_thinning	- perform morphological thinning

Noise.lib – noise generating routines

gamma_noise	- add gamma noise to an image
gaussian_noise	- add gaussian noise
neg_exp_noise	- add negative-exponential noise

rayleigh_noise	- add Rayleigh noise
speckle_noise	- add speckle (salt-and-pepper) noise
uniform_noise	- add uniform noise

Segmentation.lib – image segmentation routines

auto_threshold_segment	-perform automatic thresholding segmentation
dice_coef	-calculates Dice coefficient shape comparison metric
fuzzyc_segment	- perform Fuzzy c-Means color segmentation
gray_quant_segment	- perform gray level quantization
gvfsnake_segment	- perform gradient vector flow snake segmentation
hist_thresh_segment	- perform adaptive thresholding segmentation
jaccard_index	-calculates Jaccard index comparison metric
igs_segment	- perform improved gray scale (IGS) quantization
median_cut_segment	- perform median cut segmentation
multi_resolution_segment	- perform multiresolution segmentation
overlap_coef	-calculates Overlap coefficient comparison metric
otsu_segment	- perform Otsu method thresholding
pct_median_segment	- perform PCT/median cut segmentation
sct_split_segment	- perform SCT/Center Split segmentation
split_merge_segment	- perform split and merge, and multiresolution segmentation
threshold_segment	- perform binary threshold on an image
watershed_segment	- perform watershed segmentation on an image

SpatialFilter.lib – spatial filtering routines and noise creation

ace2_filter	- adaptive contrast filter, adapts to local gray level statistics
acuity_night_vision_filter	- visual acuity and night vision application (various blur levels)
ad_filter	- anisotropic diffusion filter, for smoothing and noise mitigation
adaptive_contrast_filter	- adaptive contrast filter, adapts to local gray level statistics
adaptive_median_filter	- median filter algorithm that retains image details
alpha_filter	- perform an alpha-trimmed mean filter
boie_cox_filter	- perform a Boie-Cox edge detection
canny_filter	- perform a Canny edge detection
cerchar_filter	-perform a Cervenka and Charvat multispectral edge detection
contra_filter	- perform a contra-harmonic mean filter
convolve_filter	- convolve an image with a matrix
edge_detect_filter	- perform edge detection on an image (Frei-Chen, Kirsch, Laplacian, Prewitt, Pyramid, Roberts, Robinson, or Sobel)

edge_link_filter	- links edge points into lines
exp_ace_filter	- adaptive contrast filter using exponential equation
geometric_filter	- performs a geometric mean filter
get_default_filter	- gets Matrix for predefined spatial masks, used with convolve_filter
harmonic_filter	- performs a harmonic mean filter
harris_corner_filter	- performs a Harris corner detection
hough_filter	- performs an Hough transform, links specified lines
improved_mmse_filter	- improved version of minimum mean squared error restoration filter
image_sharp	- sharpening algorithm II
kirsch_filter	- kirsch edge detection, returns magnitude and direction images
kuwahara_filter	- adaptive, edge-preserving, smoothing spatial filter
log_ace_filter	- adaptive contrast filter using log equation
marr_hildreth_filter	- perform Marr-Hildreth edge detection
maximum_filter	- performs a maximum filter
mean_filter	- perform a mean filter
median_filter	- performs a fast histogram-method median filter
midpoint_filter	- performs a midpoint filter
minimum_filter	- performs a minimum filter on an image
mmse_filter	- minimum mean squared error restoration filter
moravec_corner_filter	- performs Moravec corner detection
param_ad_filter	- anisotropic diffusion filter, for smoothing and noise mitigation
pratt_merit	- calculate Pratt's figure of merit
prewitt_filter	- Prewitt edge detector, returns magnitude and direction images
raster_deblur_filter	- raster deblurring filter
robinson_filter	- Robinson edge detector, returns magnitude and direction images
shen_castan_filter	- perform a Shen-Castan edge detection
single_filter	- performs geometric manipulation and enhancement with a single spatial filter
smooth_filter	- smooths the given image (kernel size in the range 2 ~ 10)
sobel_filter	- Sobel edge detector, returns magnitude and direction images
unsharp_filter	- performs unsharp masking algorithm
Ypmean_filter	- performs a Y_p mean filter

Transform.lib – two-dimensional unitary image transforms

dct_transform	- performs block-wise Discrete Cosine Transform
fft_phase	- normalizes magnitude and remaps phase data into a BYTE image
fft_transform	- performs block-wise Fast Fourier Transform
haar_transform	- performs forward or inverse Haar transform
idct_transform	- performs inverse Discrete Cosine Transform
ifft_transform	- performs inverse Fast Fourier Transform
wavtdaub4_transform	- performs wavelet transform based on Daubechies wavelet
wavthaar_transform	- performs wavelet transform based on Haar wavelet
walhad_transform	- performs Walsh/Hadamard transform (forward or inverse)

TransformFilter.lib – transform filtering routines

Butterworth_Band_Pass	- apply Butterworth bandpass filter in transform domain
Butterworth_Band_Reject	- apply Butterworth bandreject filter in transform domain
Butterworth_High	- apply Butterworth highpass filter in transform domain
Butterworth_Low	- apply Butterworth lowpass filter in transform domain
geometric_mean	- geometric mean restoration filter
High_Freq_Emphasis	- perform a high frequency emphasis (HP Butterworth + offset)
h_image	- creates an image for the degradation function, $h(r,c)$
homomorphic	- perform homomorphic filtering
Ideal_Band_Pass	- apply ideal bandpass filter in transform domain
Ideal_Band_Reject	- apply an ideal bandreject filter in transform domain
Ideal_High	- apply ideal highpass filter in transform domain
Ideal_Low	- apply ideal lowpass filter in transform domain
inverse_xformfilter	- perform inverse restoration filter
least_squares	- perform least squares restoration filter
nonfft_xformfilter	- perform standard filters (lowpass, highpass, etc.) on non-FFT symmetry transforms
notch	- perform a notch filter
parametric_wiener	- parametric wiener restoration filter, variable gamma
power_spect_eq	- power spectrum equalization restoration filter
simple_wiener	- simple wiener restoration filter (K parameter)
wiener	- wiener restoration filter