Custom Functions Summary

APPENDIX

This appendix lists by name all the custom functions used in this edition of the book. The source code for each function is in your copy of the *DIPUM3E Support Package*. Section 1.8 contains details about the package and how to obtain it. The page numbers listed in the table refer to pages where a function is first used or mentioned (see the Index for additional page references to these functions). Our use of the symbol "—" as a page reference indicates a function that is not mentioned explicitly in the book.

DIPUM3E CUSTOM FUNCTIONS TABLE

Function Name	Description	Page
`adpmedian`	Performs adaptive median filtering.	273
`aggfcn`	Aggregation function for a fuzzy system.	179
`approxfcn`	Approximation function.	181
`average`	Computes the average value of a 1-D or 2-D array.	62
`bandfilter`	Computes frequency domain band filter transfer functions.	229
`basisImages`	Displays the basis images of a transformation matrix.	465
`bayesgauss`	Classifier for Gaussian patterns.	903
`bellmf`	Bell-shaped membership function.	176
`binary2rgb`	Converts high values in a binary image to one RGB color.	—
`bound2eight`	Convert a 4-connected boundary to an 8-connected boundary.	794
`bound2four`	Convert an 8-connected boundary to a 4-connected boundary.	795
`bound2im`	Converts a boundary to an image.	790
`boundarydir`	Determines the direction of a sequence of planar points.	806
`bsubsamp`	Subsamples a boundary.	795
`changeclass`	Changes the storage class of an image.	—
`chromaticityDiagram`	Plots a chromaticity diagram.	388
`cnnactivate`	Activation function for CNNs.	956
`cnnbp`	Convolutional neural network backpropagation.	959
`cnnclassify`	Classify input images using a cnn.	961
`cnnff`	Convolutional net network feedforward.	957
`cnngradients`	Computes gradients for use in cnn weight updates.	956
`cnninfo`	Parameters and notation of convolutional neural net.	956

`cnninit`	Initializes convolutional neural network.	957
`cnnpool`	Pools (subsamples) the elements of a feature map.	956
`cnntrain`	Training of convolutional neural network.	956
`cnnupdateweights`	Updates the weights and biases of a cnn.	956
`cnotch`	Generates notch filter transfer functions.	235
`colorgrad`	Computes the vector gradient of an RGB image.	445
`colorMatchingFunctions`	Generates CIE color-matching functions.	384
`colorseg`	Performs segmentation of a color image.	448
`colorSwatches`	Displays a set of colors on individual squares.	392
`compare`	Computes and displays the error between two matrices.	520
`connectpoly`	Connects the vertices of a polygon with straight lines.	795
`conwaylaws`	Applies Conway's genetic laws to a single pixel.	607
`coord2mask`	Generates a binary mask from given coordinates.	759
`cornerprocess`	Processes the output of function `cornermetric`.	—
`covmatrix`	Computes the covariance matrix and mean vector.	848
`curveDisplay`	Display of 2-D curve.	739
`curveManualInput`	Manual input of curve coordinates.	738
`cv2tifs`	Decodes a TIFS2CV compressed image sequence.	578
`defuzzify`	Output of fuzzy system.	179
`dftfilt`	Performs frequency domain filtering.	210
`dftuv`	Computes meshgrid frequency matrices.	217
`diameter`	Measures the diameter and related properties of image boundaries.	814
`div2D`	Computes the divergence of a 2D vector field.	—
`elemdup`	Duplicates the elements of an array in specified dimensions.	—
`endpoints`	Computes end points of a binary image.	—
`fcnnactivate`	Activation function for FCNNs.	943
`fcnnbp`	Backpropagation in fully-connected neural net.	942
`fcnnclassify`	Fully-connected neural network classifier.	943
`fcnnff`	Feedforward in a fully-connected neural net.	936
`fcnninfo`	Parameters and notation of fully-connected neural net.	935
`fcnninit`	Initialize fully-connected neural net.	936
`fcnnMSE`	Outputs the mean squared error in a fully-connected neural net.	—
`fcnntrain`	Train a fully-connected neural net.	942
`fcnnupdateweights`	Updates the weights of fully-connected neural net.	942
`flipdims`	Flips an array in specified dimensions.	—
`frdescp`	Computes Fourier descriptors.	817
`freemanChainCode`	Computes the Freeman chain code of a boundary.	797
`fun2hist`	Generates a histogram from a given digital function.	123
`fuzzyfilt`	Fuzzy edge detector.	190
`fuzzysysfcn`	Fuzzy system function.	180
`fwtcompare`	Compare `wavedec2` and `wavefast`.	488
`geotrans`	Generates affine and projective geometric transformations.	330
`getCIFAR10images`	Extracts images from the CIFAR10 database.	974
`getMNISTimages`	Extracts images from the MNIST database.	969
`histroi`	Computes the histogram of an ROI in an image.	265
`hpfilter`	Computes frequency domain highpass filter transfer functions.	224
`hsi2rgb`	Converts an HSI image to RGB.	411
`huff2mat`	Decodes a Huffman encoded matrix.	536
`huffman`	Builds a variable-length Huffman code for a symbol source.	525

i2percentile	Computes a percentile given an intensity value.	662
ice	Interactive Color Editor.	428
ifrdescp	Computes inverse Fourier descriptors.	817
ifwtcompare	Compares waverec2 and waveback.	505
illuminant	Spectral power distribution of common illuminants.	—
im2jpeg	Compresses an image using a JPEG approximation.	552
im2jpeg2k	Compresses an image using a JPEG 2000 approximation.	561
im2minperpoly	Minimum perimeter polygon.	806
imageStats1	Sample function used in Chapter 2.	76
imageStats2	Sample function used in Chapter 2.	76
imageStats3	Sample function used in Chapter 2.	77
imageStats4	Sample function used in Chapter 2.	78
imageStats5	Sample function used in Chapter 2.	80
imblend	Computes the equally weighted sum of two images.	52
imcircle	Creates a binary image of circle.	708
imcolorcode	Converts values in a gray or binary image to RGB color.	712
imnoise2	Outputs noisy image and random matrix with given PDF.	255
imnoise3	Generates 2-D sinusoidal spatial patterns.	259
implfcns	Implication functions for a fuzzy system.	178
imratio	Computes the ratio of the bytes in two images/variables.	518
imstack2vectors	Extracts vectors from an image stack.	848
imtransform2	2-D image transformation with fixed output location.	—
imwarp2	Performs 2-D geometric transformation with fixed output location.	347
intensityScaling	Scale intensities of an image to the full [0 1] range.	116
intensityTransformations	Grayscale image intensity transformations.	111
interactive	Illustrates inputs from keyboard and mouse.	96
interparc	Interpolate points along a curve.	737
intline	Integer-coordinate line drawing algorithm.	795
invmoments	Computes invariant moments of an image.	843
iseven	Determines which elements of an array are even numbers.	235
isodd	Determines which elements of an array are odd numbers.	236
iswhole	True for integers (whole numbers).	58
jpeg2im	Decodes an IM2JPEG compressed image.	556
jpeg2k2im	Decodes an IM2JPEG2K compressed image.	564
kmeansClustering	Standard kmeans algorithm.	682
lambda2xyz	Converts wavelength to tristimulus values.	385
lambdafcns	Lambda functions for a set of fuzzy rules.	176
levelsetCurvature	Computes the curvature of a level set function.	757
levelsetForce	Scalar force field for level-set segmentation.	761
levelsetFunction	Generates a level-set function.	758
levelsetHeaviside	2D Heaviside and impulse for level set segmentation.	761
levelsetIterate	Iterative solution of level set equation.	755
levelsetReset	Reinitializes a signed distance function.	761
localmean	Computes an array of local means.	—
localthresh	Local thresholding.	—
lpc2mat	Decompresses a 1-D lossless predictive encoded matrix.	547
lpfilter	Computes frequency domain lowpass filter transfer functions.	220
mahalanobis	Computes the Mahalanobis distance.	892
makefuzzyedgesys	Script to make MAT-file used by FUZZYFILT.	189

`manualhist`	Generates a two-mode histogram interactively.	123
`maps2vectors`	Converts maps in the output of a cnn to vectors.	956
`mat2huff`	Huffman encodes a matrix.	532
`mat2lpc`	Compresses a matrix using 1-D lossles predictive coding.	546
`minDistanceClassifier`	Implements a minimum distance classifier.	895
`minusOne`	Multiplies an input array by (–1)^*x*+*y*.	—
`mmat2labels`	Converts a membership matrix to vector of class labels.	975
`movie2tifs`	Creates a multiframe TIFF file from a MATLAB movie.	571
`movingthresh`	Image segmentation using a moving average threshold.	—
`myRegionProps`	Properties of a single binary region.	—
`ntrop`	Computes a first-order estimate of the entropy of a matrix.	523
`onemf`	Constant membership function (one).	176
`otsuthresh`	Computes Otsu's optimum threshold from a histogram.	660
`paddedsize`	Computes padded sizes useful for FFT-based filtering.	205
`patternShuffle`	Shuffle pattern vectors.	925
`percentile2i`	Computes an intensity value given a percentile.	662
`perceptronClassify`	Perceptron classifier for two classes.	923
`perceptronTrain`	Training of two-class perceptron.	920
`pixeldup`	Duplicates pixels of an image in both directions.	—
`polyangles`	Computes internal polygon angles.	914
`predicate`	Used in function `splitmerge`.	677
`principalComponents`	Computes the principal components of a vector population.	849
`quantize`	Quantizes the elements of a UINT8 matrix.	550
`randvertex`	Adds random noise to the vertices of a polygon.	914
`recnotch`	Generates axes notch filter transfer functions.	238
`regiongrow`	Performs image segmentation using region growing.	673
`reprotate`	Rotates an image repeatedly.	—
`rgb2hsi`	Converts an RGB image to HSI.	410
`rgbcube`	Displays an RGB cube on the MATLAB desktop.	391
`rot180`	Rotates an input matrix by 180 degrees.	—
`rspd2xyz`	Converts relative spectral power density to XYZ.	385
`seq2tifs`	Creates a multi-frame TIFF file from a MATLAB sequence.	571
`showmo`	Displays the motion vectors of a compressed image sequence.	578
`sigmamf`	Sigma membership function.	176
`signature`	Computes the signature of a boundary.	809
`sinfun1`	Sample function used in Chapter 2.	68
`sinfun2`	Sample function used in Chapter 2.	70
`sinfun3`	Sample function used in Chapter 2.	71
`smf`	S-shaped membership function.	176
`snakeForce`	Components of external force for use in the snake algorithm.	736
`snakeIterate`	Iterative solution of the snake equation.	732
`snakeMap`	Computes an edge map for use in the snake iterative algorithm.	733
`snakeRespace`	Respaces the coordinates of a snake uniformly.	737
`spectrumBar`	Adds visible light spectrum bar to plot.	379
`spectrumColors`	RGB colors corresponding to the visible light spectrum.	378
`specxture`	Computes spectral texture of an image.	840
`spfilt`	Performs linear and nonlinear spatial filtering.	268
`splitmerge`	Segments an image using a split-and-merge algorithm.	677
`statmoments`	Computes statistical central moments of image histogram.	263

`statxture`	Computes statistical measures of texture in an image.	830
`strsimilarity`	Computes a similarity measure between two character vectors.	912
`subim`	Extracts a subimage, *s*, from a given image, *f*.	66
`tifs2cv`	Compresses a multi-frame TIFF image sequence.	576
`tifs2movie`	Create a MATLAB movie from a multiframe TIFF file.	571
`tifs2seq`	Create a MATLAB sequence from a multi-frame TIFF file.	570
`tofloat`	Convert image to floating point.	35
`trainingMSE`	Computes the mean squared error per epoch of training.	—
`trapezmf`	Trapezoidal membership function.	176
`triangmf`	Triangular membership function	175
`truncgaussmf`	Truncated Gaussian membership function.	176
`twodsin1`	Sample function used in Chapter 2.	70
`twodsin2`	Sample function used in Chapter 2.	72
`twodsin3`	Sample function used in Chapter 2.	73
`twomodegauss`	Generates a two-mode Gaussian function.	—
`unravel.c`	Decodes a variable length coded bit sequence.	538
`unravel.m`	Decodes a variable-length bit stream.	540
`uppermostLeftmost`	Finds the uppermost, leftmost point of a closed boundary.	791
`vectors2maps`	Converts vectors to the format of cnn output maps.	956
`visgeotrans`	Visualize geometric transformation.	—
`waveback`	Computes inverse FWTs for multi-level decomposition [C,S].	502
`wavecopy`	Fetches coefficients of a wavelet decomposition structure.	495
`wavecut`	Zeroes coefficients in a wavelet decomposition structure.	495
`wavedisplay`	Displays wavelet decomposition coefficients.	497
`wavefast`	Computes the FWT of a 3-D extended 2-D array.	484
`wavefilter`	Creates wavelet decomposition and reconstruction filters.	481
`wavepaste`	Puts coefficients in a wavelet decomposition structure.	496
`wavework`	Used to edit wavelet decomposition structures.	492
`wavezero`	Zeroes wavelet transform detail coefficients.	507
`whtmtx`	Generates a sequency-ordered Walsh-Hadamard transformation matrix.	468
`x2majoraxis`	Aligns coordinate *x* with the major axis of a region.	815
`xyy2xyz`	Converts chromaticity coordinates to XYZ tristimulus values	390
`xyz2xyy`	Converts XYZ tristimulus values to chromaticity coordinates.	388
`zeromf`	Constant membership function (zero).	175