() parentheses in MAX expressions, 112
:= (assignment operator) in MAX, 110;
(semicolon) in MAX, 109
<< (input operator) in MAX, 109
>> (output operator) in MAX, 109
4-distance, 211
4-neighbors, 86
4-regions, 86
8-distance, 211, 300
8-neighbors, 86
8-regions, 86

A
a posteriori method, 374
acceptability, 312, 373
activation function, 364–365
activation values (nodes), 364–365
acute angle emphasis, 215
Adaboost (Adaptive boosting), 317
adaptive algorithm, 327
adaptive gradient method, 51
AIPCV library, interfacing with, 14–18
algorithms
adaptive algorithm, 327
Algorithms for Image Processing and
Computer Vision website, 02
Baird algorithm, 341
Choi/Lâm/Siu algorithm, 224–226
contour-based thinning algorithms, 221–226
DFT algorithms, 268
Differential Box Counting (DBC)
algorithm, 199
ELT algorithm, 160
Fletcher algorithm, 380
median cut algorithm, 203
Otsu’s Grey Level Histogram (GLH)
algorithm, 149
popularity algorithm, 202–204
Sobel algorithm, 56
sorting algorithms, 288
Stentiford thinning algorithm, 212–215
vision algorithms, 288
Zhang-Suen algorithm, 217–220
Zhang-Suen/Stentiford/Holt
combined algorithm (source
code), 235–246
align function, 267
ALOI (Amsterdam Library of Object
Images), 397
AND elements, 365–366
angular regions, 412–413
API (application program interface), 1
arcing (boosting), 316
artifacts, skeleton, 215
artificial blur, creating, 264–269
artificial neural systems (ANS), 363–364
artificial textures, 178
ASCII code, 324
aspect ratios, 446
assignment operator (=) in MAX, 110
autosize property, 7
average keyword, 188

B
backpropagation net for digit recognition, 368–372
bagging, 315–316
Baird algorithm, 341
band-limited Laplacian, 51
band-pass/band-stop filters, 280
Berkeley Image Segmentation Dataset, 204
between-classes variances, 141
bi-level images, 137
bimodal histograms, 153
bin_ero C function, 98, 102
binary Laplacian images (BLI), 51
binary operations
  binary dilation, 88–92
  binary dilation implementation, 92–94
  binary erosion, 94–100
  binary erosion implementation, 100–101
  binary morphological operators, 87
  conditional dilation, 116–119
  counting regions, 119–121
  hit-and-miss transform operator, 113–115
MAX programming language, 107–113
  opening and closing operations, 101–107
  region boundaries, identifying, 116
binding names to targets, 448–449
black and white images, 137
black and white photographs, 399
Black scheme, 374
blobworld scheme, 205
blocking send, 434
blur, artificial, 264–269
Boolean edge density, 410–411
boosting (arcing), 316–317
bootstrap aggregation, 315–316
bootstrap samples, 315–316
Borda count, 313–314, 416
Borda method, 374
boundaries
  content-based searching and, 418
  between objects, 409–410
  boustrophedon scanning, 167
break cost, 330–331

C
cameras (webcams), 10–11
Canny, John, 42
Canny edge detector
  fundamentals of, 42–48
  source code, 62–70
Canny/Shen-Castan comparison, 51–53
capturing images, 10–13
centroid, defined, 304
chain coding, 23, 221
characters
  character outlines, properties of, 349–353
  handprinted. See handprinted characters
Choi/Lam/Siu algorithm, 224–226
Chow-Kaneko method, 152–156
chrominance, pixel, 407
circular regions, 414
circularity, 337
city block distance, 300
classifications
  bagging and boosting, 315–317
  classifiers, multiple (OCR), 372–375
classifying vegetables (example), 293
cross validation, 304–306
in-class and out-class, 295–299
minimum distance classifiers, 299–304
multiple classifiers - ensembles, 309–315
nearest neighbor classifier, 302–303
objects/patterns/statistics, 285–299
support vector machines (SVM), 306–309
visual, 295
clipping and viewing geometry (Open GL), 447
cluster-based thresholds, 170–171
collections of images, maintaining, 396–398
color
color edges, 53–58
color morphology, 131–132
color quad tree, 400
color quantization, 202
color segmentation, 201–205
color textures, 205
coordinates, 177
current colors, 446
prototype colors, 403–404
references (bibliography), 206–208
saturation (S), 56
segmentation, 178
website files, 205–206
color image features
color quad tree, 400
color-based methods, 407–408
comparing histograms, 402–403
hue and intensity histograms, 401–402
mean feature, 400
overview, 399–400
requantization, 403–404
results for searching experiments using, 404–407
complements of sets, 89
complex numbers, 254
compute_adaptive_gradient function, 51
computer networks, 440–443
computer vision, 285–287
computers, vector, 426
conditional dilation, 116–119
Condorcet winner criterion, 314
confusion matrix, 303, 349
connected regions, 86
connectedness (digital morphology), 86–87
connectivity numbers, 212–214
content-based searching
content-based image retrieval
(content-based image retrieval (CBIR), 396
data sets and, 418–419
objects/contours/boundaries, 418
query by example features. See query by example (QBE)
references (bibliography), 420–424
searching images, 395–396
spatial considerations. See spatial considerations
texture and, 418
website files, 419–420
countours
content-based searching and, 418
contour-based thinning algorithms, 221–226
contrast estimate, 185
contrast keyword, 188
convex deficiencies (OCR), 353–357
convexity of objects, 338
convolution masks, 192–193, 458
convolution of images, 253–254
Copeland rule, 315
core pixels, 328
Corel data set, 415–417
corners, wave, 210
counting regions, 119–121
covariance matrix, 301
CPU systems, 425
critical section code, 427
cross validation, 304–306
cumulative histograms, 403
current colors, 446
curvature of surfaces, 195–198
cvCaptureFromCAM function, 10–11
cvCvtColor function, 402
cvDFT function, 263
cvGet2D function, 6
cvMat function, 263
cvMatToImage function, 269
cvMoveWindow function, 7
cvNamedWindow function, 7
cvScalar function, 6, 402
cvSet1D and cvSet2D functions, 263
cvShowImage function, 7

data, training, 294–295
data sets, content-based query systems
and, 418–419
decision trees, 331–332
deconvolving images, 252
degradation of images, 251–253
density, edge, 409–410
depth field (IplImage), 4
derivative operators, 30–35
descriptors
defined, 183
results from GLCM, 186
DFT algorithms, 268
Diff array, 258
difference histograms, 186–187
Differential Box Counting (DBC) algorithm, 199
digit recognition
applications in, 358
backpropagation net for, 368–372
digital bands, defined, 229–230
digital Laplacian, 139
digital morphology
binary operations. See binary operations
color morphology, 131–132
connectedness, 86–87
grey-level morphology. See grey-level morphology
morphology defined, 85–86
references (bibliography), 135–136
website files, 132–135
dilation
binary, 88–94
conditional, 116–119
operations, defined, 85
discrete Fourier transform (DFT), 255
discrete inverse Fourier transform, 260
dispersion, vector, 193–195
displaying images, 7–10
dissenting-weighted majority vote (DWMV), 311, 373
distance
4-distance, 211
8-distance, 300
city block, 300
distance maps, 105
Euclidean, 300
between features, 302–304
Mahalanobis, 300–302
Manhattan, 300
metrics, 300–302
Pythagorean, 300
distributed computing, 426
do expression statements (MAX), 109
domains, defined (objects), 288
downloading software, 460–461
dxy_seperable_convolution C function, 45
digit recognition
Canny edge detector, 42–48
Canny edge detector C program source code, 62–70
Canny/Shen-Castan comparison, 51–53
color edges, 53–58
defined, 22
derivative operators, 30–35
Marr-Hildreth edge detector, 39–42
Marr-Hildreth edge detector source code, 58–61
models of edges, 24–26
noise, 26–30
purpose of, 21–23
references (bibliography), 82–84
Shen-Castan (ISEF) edge detector, 48–51
Shen-Castan edge detector source code, 70–80
Sobel edge detector, 36
template-based, 36–38
theory and traditional approaches, 23
website files, 80–82
edges
density, 409–410
direction, 410–411
enhancement, defined, 22
linking, 345
magnitude, 31
pixels, 139–140, 410
response, 31
tracing, defined, 23
edge-level thresholding (ELT). See
ELT (edge-level thresholding)
ensuring results from
cozoccurrence matrices with, 190
modeling illumination using, 156–159
ramp edges, 23–24
texture and, 188–191
use of in OCR of faxed images, 345–348
elimination processes, 314
elliptic points, 197
ELT (edge-level thresholding)
algorithm, comparison with other
thresholding methods, 160
defined, 158
implementation and results, 159–160
in poor illumination situations, 160
endpoints, 212–213
energy, texture and, 191–193
ensemble classifiers, 309
entropy
calculating, 186
using, 142–145
erosion
binary, 94–101
erosion-dilation duality, proof of, 98
operations, defined, 85
error rate (edge detection), 42–43
Euclidean distance, 211, 300
Euler number, 338
evenodd function, 258
execution timing
clock() function, 428–430
overview, 427–428
QueryPerformanceCounter, 430–432
F1 measure, 406
face image example, 149, 151, 156–157,
163, 168, 171–172
false positives/negatives (edge
detection), 33
false zero-crossing suppression, 51
fast Fourier transform (FFT), 256–259
fax images, OCR on. See OCR on fax
images
features
for classifying vegetables, 293
color image. See color image features
distance between, 302–304
for query by example. See query by
eexample (QBE)
and regions, 288–292
fftImage function, 267–268
fftlib.c procedures, 273
filtering
band-pass/band-stop filters, 280
frequency domain filters, 280
frequency filters, 278–280
high-emphasis filters, 280
high-pass filters, 279
Homomorphic filtering, 277–281
inverse filter, 270–271
kfill filters, 328–329
low-pass filters, 279–280
median filters, 327, 437
notch filters, 275
Wiener filter, 271–272
fixed-size images, using as templates, 419
flag parameter, 262
Fletcher algorithm, 380
force fields, use of, 230–234
force-based thinning
digital bands, defined, 229–230
force fields, use of, 230–234
force-based thinning (continued)
overview, 228–229
segments, digital band, 230
skeletons of stubs, 230
stubs, defined, 230
subpixel skeletons, 234–235
FORTRAN language, 154
Fourier domain, 253
Fourier transforms
defined, 253
discrete Fourier transform (DFT), 255
fast Fourier transform, 256–259
fundamentals, 254–256
inverse Fourier transform, 260
in OpenCV, 262–264
two-dimensional Fourier transforms, 260–262
fractal dimension, 198–201
fragment and vertex shaders, 452–453
frequencies
frequency filters, 278–280
spatial frequencies, 278–279
frequency domain
artificial blur, creating, 264–269
basics, 253–254
fast Fourier transform, 256–259
filters, 280
Fourier transform, 254–256
Fourier transforms in OpenCV, 262–264
inverse Fourier transform, 260
two-dimensional Fourier transforms, 260–262
fromOpenCV function, 16
F-score, 406
fuzzy sets, 146–148

G
Gaussian
curves, 139, 197
filter mask, 45
noise, 29, 43
smoothing filter, 39–40
GLEW utility, 458, 461
globally eroded images, 105
GLSL (OpenGL Shading Language)
basics, 444–445
required initializations of, 453–454
GLUT, 461
glyphs
defined, 322
glyph boundaries, vectorizing, 346
isolating individual (scanned OCR), 329–333
GPU (graphics processing unit)
developing/testing shader code, 459–460
GLSL (OpenGL Shading Language), 444–445
OpenGL background and fundamentals, 445–447
overview, 444
practical textures in OpenGL, 448–451
programming example, 457–458
reading/converting images, 454–455
shader programming basics, 451–454
shader programs, passing parameters to, 456–457
speedup with, 459
gradients
morphological (grey-level), 128
multi-dimensional, 53
Graphics Gems, 228
graphs
graph grammars, 382
graph parsers, 382
of processing elements, 365
grey histograms, 409
grey level co-occurrence matrix (GLCM)
contrast and, 185
descriptors, results from, 186
entropy, calculating, 186
fundamentals of, 183–184
homogeneity and, 185
maximum probability entry, 185
moments and, 185
texture operators, speeding up, 186–188
grey levels, 26–28
grey sigma, 409
grey-level histograms method, 141–142
grey-level images
analysis of texture in, 179–182
code for writing, 7
features, 408–411
grey-level morphology
elementary, 125
fundamentals of, 121–123
morphological gradient, 128
opening/closing grey-scale images, 123–126
segmentation of textures, 129–130
size distribution of objects, 130–131
smoothing operations, 126–127
grey-level segmentation. See also thresholding
cluster-based thresholds, 170–171
texture operators, speeding up, 186–188
grey levels, 26–28
grey sigma, 409
grey-level histograms method, 141–142
grey-level images
analysis of texture in, 179–182
code for writing, 7
features, 408–411
grey-level morphology
elementary, 125
fundamentals of, 121–123
morphological gradient, 128
opening/closing grey-scale images, 123–126
segmentation of textures, 129–130
size distribution of objects, 130–131
smoothing operations, 126–127
grey-level segmentation. See also thresholding
cluster-based thresholds, 170–171
texture operators, speeding up, 186–188
grey levels, 26–28
grey sigma, 409
grey-level histograms method, 141–142
grey-level images
analysis of texture in, 179–182
code for writing, 7
features, 408–411
grey-level morphology
elementary, 125
fundamentals of, 121–123
morphological gradient, 128
opening/closing grey-scale images, 123–126
segmentation of textures, 129–130
size distribution of objects, 130–131
smoothing operations, 126–127
grey-level segmentation. See also thresholding
cluster-based thresholds, 170–171
texture operators, speeding up, 186–188
grey levels, 26–28
grey sigma, 409
grey-level histograms method, 141–142
grey-level images
analysis of texture in, 179–182
code for writing, 7
features, 408–411
grey-level morphology
elementary, 125
fundamentals of, 121–123
morphological gradient, 128
opening/closing grey-scale images, 123–126
segmentation of textures, 129–130
size distribution of objects, 130–131
smoothing operations, 126–127
grey-level segmentation. See also thresholding
cluster-based thresholds, 170–171
texture operators, speeding up, 186–188
grey levels, 26–28
grey sigma, 409
grey-level histograms method, 141–142
grey-level images
analysis of texture in, 179–182
code for writing, 7
features, 408–411
H
hairs (artifacts), 215
handprinted characters
character outline, properties of, 349–353
convex deficiencies, 353–357
neural nets, 363–372
overview, 348–349
vector templates, 357–363
Hare, Thomas, 314
height field (IplImage), 3
Height parameter, 456–457
hex feature, 401
hidden layers (processing elements), 367
hierarchical template matching, 336
high-emphasis filters, 280
highgui library, 7
high-pass filters, 279
high-performance computing
CPU systems, 425
execution timing. See execution timing
GLSL, required initializations of, 453–454
GPU. See GPU (graphics processing unit)
message passing, 427
Message-Passing Interface (MPI) system. See Message-Passing Interface (MPI) system
multiple-processor computation, paradigms for, 426–427
references (bibliography), 461–463
shared memory, 426–427
website files, 461
histograms
comparing, 402–403
grey, 409
hue and intensity, 401–402
slope, 338
source code for calculating sum and difference, 189
sum, 186–187
hit-and-miss transform operator, 113–115
holes in objects, 338
Holt variation of Zhang-Suen, 218–221
homo keyword, 188
homogeneity, 185
homomorphic filtering, 277–281
horizontal projections, 376
Hough image, 343
Hough space, 342–343
Hough transforms, 253, 342–344, 377
Hubble Space Telescope example, 252
hue (color edges), 53, 56
hue and intensity histograms, 401–402
Hurst coefficient, 199–201
hybrid regions, 414
hysteresis thresholding, 48

I
ideal step edge, 23, 25
if (expression) then statements (MAX), 109
illumination
effects, isolating, 280–281
modeling using edges, 156–159
images
capturing, 10–13
color feature. See color image features
deconvolving, 252
degradation of, 251–253
displaying, 7
image degradations, 251–253
image generator, MAX, 112
image processing, 285
IMAGE variable type, 108
image-analysis software, 1–2
imageData field (IplImage), 3
imageDataOrigin field (IplImage), 4
image-processing tasks, 425
imageSize field (IplImage), 4
img variable, 7
maintaining collections of, 396–398
monochrome, 137
OCR on simple perfect images, 322–326
reading/converting, 454–455
reading/writing, 6–7
restoration of. See restoration of images
scan lines in, 126
searching, 395–396
indirect access (accessing pixels), 6
infinite symmetric exponential filter (ISEF), 49
initializations, required GLSL, 453–454
input (<<) operator (MAX), 109
inputs (processing elements), 364–365
installing MPI, 432–433
INT variable type, 108
inter-process communication, 434–435
intersections of sets, 89
inverse
filter, 270–271
Fourier transform, 260
moments, 185
IplImage
converting to AIPCV from, 15–16
data structure, 3–6
Iris data set (classification example), 296–299, 302, 305–306
is_candidate_edge function (ISEF code), 51
ISEF (infinite symmetric exponential filter), 49
iterative morphological methods, 212–221
iterative selection, 140–141, 152

J
JPEG files, noise and, 271

K
Kapur’s method, 144
kernels, defined, 308
kFill filter, 328–329
Kirsch operator, 37–38
Kitchen and Rosenfeld, 33
Kittler and Illingworth function, 148–149
k-nearest neighbor method, 303–304
kurtosis, 181–182
L
Laplacian
digital, 139
differential operator, 39
digitization, 191
LARGE_INTEGER structure, 430
leave-one-out cross validation, 306
level of detail parameter, 449
LIBSVM software, 309
line adjacency graphs (LAGs), 378–379, 381
line fuzz (artifacts), 215, 217
line of symmetry, 210
linear discriminants, 299
linking, edge, 345
local edge coherence, 33
localization (edge detection), 42–43
loop ... end statements (MAX), 109
low-pass filters, 279
LPBoost scheme, 317

M
Mahalanobis distance, 300–302
majority criterion, 314
Manhattan distance, 300
mapping, texture (OpenGL), 450–451
margins, 307
Marr, David, 39
Marr-Hildreth edge detector
fundamentals of, 39–42
source code, 58–61
MARS (Multimedia Analysis and Retrieval System), 418
masking, unsharp, 128
masks, Sobel, 410
matching templates (scanned OCR), 325, 329–333
MAX (Morphology And eXperimentation) programming language, 107–113
maximum probability entries, 185
mean
feature, 400
grey level, 180–181
medial axis function (MAF), 210–212
median
cut algorithm, 203
filters, 327, 437
membership function, 146
memory, shared, 426–427
merging
multiple classifiers, 372–374
multiple methods, 309–310
type 1 responses, 310–311
type 2 responses, 313–315
type 3 responses, 315
Message-Passing Interface (MPI) system
installing, 432–433
inter-process communication,
434–435
network cluster computing, 440–443
overview, 432
programs, running, 436–437
real image computations, 437–440
using, 433–434
messages
message expression statements (MAX), 110
passing, 427
sending/receiving, 434
methods, color-based, 407–408
metrics, distance, 300–302
minimum distance classifiers
distance between features, 302–304
distance metrics, 300–302
overview, 299
minimum error thresholding, 148–149
MLS (moving least-squares), 158, 160
models of edges, 24–26
moments, statistical, 181, 185, 338, 407–408
monochrome images, 137
monotonicity criterion, 314
Moore, Gordon E., 425
morphology
defined, 85–86
digital. See digital morphology
morphology (continued)
morphological boundary extraction, 116
morphological gradient (grey-level), 128
morphological operators, binary, 87
motion blur, 276–277
moving averages, 167–169
moving least-squares (MLS) scheme, 158
moving weighted average method, 158
MPI (Message-Passing Interface) system. See Message-Passing Interface (MPI) system
multiple classifiers
ensemble classifiers, 309
evaluation process, 311–312
merging multiple methods, 309–310
merging type 1 responses, 310–311
merging type 2 responses, 313–315
merging type 3 responses, 315
response types, converting between, 312–313
use of, 372–375
multiple CPU systems, 425
multiple features, 338
multiple methods, merging, 309–310
multiple pixel concept, 222
multiple thresholds, 171–172
multiple-processer computation, paradigms for, 426–427
music symbol recognition, 381–382
neural nets
advantages of, 363–364
backpropagation net for digit recognition, 368–372
simple neural net example, 364–368
nodal pixels, 106
noise
defined, 23–24
fundamentals of, 26–30
reduction of (scanned OCR), 327–329
structured, 273–275
nonmax_suppress C function, 47
non-maximum suppression, 45–46
notch filters, 275
Numerical Recipes in C, 154
n-way cross validation, 305

O
objects
boundaries between, 409–410
content-based retrieval and, 418
defined, 287
recognition of, 287–288
size distribution of (grey-level), 130–131
treating as polygons, 226–228
vision systems looking for, 288
OCR (optical character recognition) on fax images. See OCR on fax images
problem of, 321–322
on scanned images. See OCR on scanned images
on simple perfect images, 322–326
OCR on fax images
edges, use of, 345–348
overview, 339
skew detection, 340–344
OCR on scanned images
isolating individual glyphs, 329–333
noise reduction, 327–329
overview, 326–327
statistical recognition, 337–339
template matching, 329–333

N
naturally occurring textures, 178
nChannels field (IplImage), 4
nearest centroid method, 304
nearest neighbor classifier, 302–303
necking artifact (skeletons), 215, 217
negative zero crossing, 51
neighborhood (pixels), 328
network cluster computing, 440–443
networks of processing elements, 365
neural net recognition system (source code), 383–390
OMR (optimal music recognition), 375–376. See also printed music recognition
one dimensional recursive filters, 49–50
open direction, defined, 354
OpenCV system
   basic code, 2–10
   converting AIPCV images to, 14–15
   displaying images, 7
   Fourier transforms in, 262–264
   IplImage data structure, 3–6
   program to read/process/display images (example), 7–10
   reading images from files with, 455
   reading/writing images, 6–7
   versions and companion tools, 2
OpenGL
   background and fundamentals, 445–447
   textures in, 448–451
   websites for downloading/documentation, 461
opening/closing grey-scale images, 123–126
opening/closing operations, 101–107
operators
   for locating edges, 29–30
   MAX, 110–112
optical character recognition (OCR). See OCR (optical character recognition)
OR function (XOR), 366–367
OR operator, 262
origin field (IplImage), 4
origin-centered Fourier transforms, 270–271
Otsu’s Grey Level Histogram (GLH) algorithm, 149
output (>>) operator (MAX), 109
output function, 364–365
output values (processing elements), 364–365
overall regions, 411

P
parabolic points, 197
paradigms for multiple-processor computation, 426–427
parallel computers, 426–427
parallel method, defined, 218
parameters
   passing to shader programs, 456–457
   texture, 449–450
parliamentary majority vote, 310
parsers, graph, 382
partners, code, 438
pascal image example, 149–150, 152, 156–157, 163, 168, 171–172
passing messages, 427
patterns over a region (texture), 177, 287
pixel masks, 191
pixel representations for RGB images, 4–6
PIXEL variable type, 108
pixels, edge, 139–140
pmax keyword, 188
point spread function (PSF), 252
polygons
   drawing (OpenGL), 448
   treating objects as, 226–228
popularity algorithm, 202–204
positive zero crossing, 51
precision (information retrieval), 405–406
primal sketch, 39
printed music recognition
   music symbol recognition, 381–382
OMR (optimal music recognition)
   overview, 375–376
   segmentation and, 378–380
   staff lines, 376–378
processing elements (PEs), 364
profiles, 338, 350
program objects, 454
PROJECTION mode, 446
projections, horizontal, 323
properties of character outlines, 349–353
proportional spacing (text), 327
proto feature, 404
prototype colors, 403–404
p-tile method, 137
Pythagorean distance, 300

Q
QBIC (Querying Images by Content), 418
quad feature, 401
quad trees, 400–401
quantization
  requantization, 403–404
  uniform, 202–203
query by example (QBE)
  color image features. See color image
  features
example, 399
  grey-level image features, 408–411
  overview, 399

R
radial basis functions, 308
ramp edge, 23–24
raster images
  converting into vector templates, 359
  representing objects with, 286
reading/converting images, 454–455
reading/writing images, 6–10
recall (information retrieval), 406
recognition
  of objects, 287–288
  rates, 351–353, 356–357
  reliability, 312
rectangular regions, 412
rectangularity, 337
recursive filters, 49–50
reduction, color, 399
references (bibliography)
  classification, 318–319
  content-based searching, 420–424
  digital edge morphology, 135–136
  edge detection, 82–84
  grey-level segmentation, 173–175
high-performance computing, 461–463
restoration of images, 283–284
symbol recognition, 392–394
texture and color, 206–208
thinning, 247–249
vision system practical aspects, 18–19
reflections of sets, 89
regional thresholds
  Chow-Kaneko method, 152–156
  ELT algorithm, comparison with
    other thresholding methods, 160
  ELT thresholding implementation
    and results, 159–160
  modeling illumination using edges,
    156–159
  overview of, 151–152
regions
  connected, 86
  counting, 119–121
  features and, 288–292
  identifying boundaries of, 116
  rejections (classification), 349
  relaxation methods, 161–167
  reliability formula, 311–312
  render function, 458–459
  rendering images, 286
  requantization, 403–404
  response (edge detection), 42
  response types, converting between
    multiple classifiers, 312–313
restoration of images
  frequency domain. See frequency
  domain
  Homomorphic filtering, 277–281
  illumination effects, isolating,
    280–281
image degradations, 251–253
inverse filter, 270–271
motion blur, 276–277
references (bibliography), 283–284
structured noise, 273–275
website files, 281–282
Wiener filter, 271–272
RGB images
   code for writing, 7
   pixel representations for, 4–6
RGB values, 56
   RGB/RGBA formats, 455
roi field (IplImage), 4
Rosenfeld and Kitchen, 33
rotations, defined, 253
roughness spectrum, 106–107

S
saddle points, 197
sampler, defined (texture), 457
saturation (S), color, 56
scan lines in images, 126
scanned images, OCR on. See OCR on
   scanned images
scattergrams, 292
search engine evaluation scheme, 406
search sets, 399
searching images, 395–396
segmentation
   color, 201–205
   defined, 21
   in printed music recognition, 378–380
   texture and, 177–179
   of textures (grey-level), 129–130
segments, digital band, 230
separable convolution C function, 45
shader programming
   basics, 451–454
   passing parameters to programs, 456–457
   shader code, developing/testing, 459–460
   ShaderDesigner tool (Typhoon Labs), 459
Shannon's function, 147
shape numbers, 338
shared memory systems, 426–427, 444
Shen-Castan edge detector
   fundamentals of, 48–51
   to locate pixels belonging to object boundaries, 157
Shen-Castan/Canny comparison, 51–53
   source code, 70–80
sigma, grey, 409
signal-dependent noise, 29
signal-independent noise, 26
signatures, defined, 338–339
signed sequential Euclidean distance (SSED) transform, 225–226
simple majority vote (SMV), 310, 372
single threshold selection, sample results from, 149–151
size distribution of objects (grey-level), 130–131
skeletons
   basics, 209
   of stubs, 230
   subpixel, 234–235
skew angles, 340–341
skew detection (OCR), 340–344
skewness, 181–182
sky image example, 149–150, 152, 156, 163, 168, 171–172
slave processors, 440–443
slope histograms, 338, 346–347
slow4 program, 259
smallest standard deviation, 192
smoothing operations (grey-level), 126–127
Sobel
   algorithm, 56
   edge detector, 36, 191
   masks, 410
software, downloading required, 460–461
sorting algorithms, 288
source code
   for calculating sum and difference histograms, 189
   Canny edge detector, 62–70
   Marr-Hildreth edge detector, 58–61
   neural net recognition system, 383–390
   Shen-Castan edge detector, 70–80
   Zhang-Suen/Stentiford/Holt combined algorithm, 235–246
spatial considerations
  angular regions, 412–413
  circular regions, 414
  hybrid regions, 414
  overall regions, 411
  overview, 411
  rectangular regions, 412
  spatial frequencies, 278–279
  test of spatial sampling, 414–417
  speed values, 276
  spikes (bright spots), 273–274
  spurious projections (artifacts), 215, 217
SSED transform, 225–226
staff lines (music OCR), 376–378
staircases, 25–26
standard deviation, 27, 29, 181, 301
  statistical
    moments, 181
    pattern recognition, 288
    recognition (scanned OCR), 337–339
stddev keyword, 188
steepest descent method, 370
Stentiford thinning algorithm, 212–213, 215
step edges, 23–25
strings providing image path name, 3
structural pattern recognition, 337
structured noise, 273–275
structuring elements, 89
stubs
  defined, 230
  skeletons of, 230
subpixel skeletons, 234–235
sub-regions, types of, 411
success rates, 288, 405
Sum array, 258
sum histograms, 186–187
support vector machines (SVM), 306–309
surfaces
  curvature of, 195–198
  texture and, 193–198
SVMlight, 309
symbol recognition

handprinted characters. See
  handprinted characters
multiple classifiers, 372–375
neural net recognition system
  (source code), 383–390
optical character recognition. See
  OCR (optical character recognition)
printed music recognition. See
  printed music recognition
references (bibliography), 392–394
website files, 390–392

T
tailing (artifacts), 215, 217, 223
Tamura features (texture), 418
targets
  binding names to, 448–449
  defined (objects), 289
templates
  template matching (scanned OCR), 325, 329–333
  template-based edge detection, 36–38
  using fixed-size images as, 419
vector (OCR), 357–363
vector template style of match, 348
testing
  shader code, 459–460
  spatial sampling, 414–417
  training and, 292–295
textons, 177
textures
  analysis of texture in grey-level images, 179–182
  artificial, 178
  color textures, 205
  content-based searching and, 418
  edges and, 188–191
  energy and, 191–193
  fractal dimension, 198–201
  grey-level co-occurrence and. See
    grey level co-occurrence matrix (GLCM)
in OpenGL, 448–451
  operators, speeding up, 186–188
references (bibliography), 206–208
segmentation and, 129–130, 177–179
surfaces and, 193–198
texture lookup function, 457
website files, 205–206

thinning
approaches to, 210
Choi/Lam/Siu algorithm, 224–226
contour-based thinning algorithms, 221–226
defined, 209–210
force-based thinning. See force-based thinning
iterative morphological methods, 212–221
medial axis function (MAF), 210–212
references (bibliography), 247–249
skeletons, 209–210
treating objects as polygons, 226–228
triangulation methods, 227–228
website files, 246
Zhang-Suen/Stentiford/Holt combined algorithm (source code), 235–246

threshold_edges C function, 51

thresholding
cluster-based thresholds, 170–171
ELT thresholding implementation and results, 159–160
hysteresis, 48
minimum error, 148–149
multiple thresholds, 171–172
single threshold selection, sample results from, 149–151
thresholding images (example), 7–10
TIFF files, saving images as, 271
timing, execution. See execution timing
toOpenCV function, 16

tracers, 223
training, testing and, 292–295
transform operator, hit-and-miss, 113–115
transforms, defined, 253
translations of sets, 88
triangles, drawing with OpenGL, 441
triangulation methods, 227–228

trivial regions, 411
Tsallis entropy, 144
two-dimensional Fourier transforms, 260–262, 425
type 1, 2, 3 responses, 309–315

U
uchar (unsigned character), 5
unary operators, 112
uniform quantization, 202–203
variables, 456
union of sets, 89
unsharp masking, 128, 458
unsigned int/unsigned long, 428

V
value (V), color, 56
vectors
support, 307–308
vector computers, 426
vector dispersion, 193–195
vector template style of match, 348
vector templates (OCR), 357–363
vector-dispersion method, 195
vectorization, 345–346
vegetable classification example, 293
vertex and fragment shaders, 451–453
viewing direction, 447

vision
algorithms, 288
systems, 288
classification, 295

W
wavelet, defined, 401
webcams, capturing images with, 10–13
website files
classification, 317–318
content-based searching, 419–420
digital morphometry, 132–135
digitization, 80–82
grey-level segmentation, 172–173
website files (continued)
  high-performance computing, 461
  restoration of images, 281–282
  symbol recognition, 390–392
  texture and color, 205–206
  thinning, 246

websites, for downloading
  ALOI (Amsterdam Library of Object Images), 397
  code and data for this book, 18
  GLEW, 461
  GLUT, 461
  MPI, 432, 460
  OpenCV versions 1.1 and 2.0, 2
  OpenGL, 461
  ShaderDesigner tool (Typhoon Labs), 459

websites, for further information
  Adaboost (Adaptive boosting), 317
  LIBSVM, 309
  support vectors, 309
  SVMlight, 309
  WEKA system, 309
  weight values (processing elements), 364

weighted averaging, 159
weighted majority vote (WMV), 310, 373
WEKA system, 309
white Gaussian noise, 43
width field (IplImage), 3
Width parameter, 456–457
widthStep field (IplImage), 4
Wiener filter, 271–272
Windows Command Prompt program, 436
within-class variances, 141
wmpiregister.exe MPI program, 441
writing/reading images, 6–10

X
  X functions (IPCV library), 17–18
  XOR (OR function), 366–367

Z
  zero crossings, 39, 232
  Zhang-Suen algorithm, 217–220
  Zhang-Suen/Stentiford/Holt combined algorithm (source code), 235–246