Index

A

- A streams in multistreaming, 76–78, 84, 89
- access to GPU memory, 527–528
- accessing octrees, 599–601
- accumulated energy values in progressive refinement, 637–638
- accumulation in antialiased rasterization, 339
- acquisition of materials, 179–181
- active levels for geometry clipmaps, 32
- active tiles, 539
- adaptive sampling, 279–280
- adaptive subdivision, 112
  - in global illumination, 643–645
  - surfaces in, 584
  - texture quadtrees in, 644
- adaptive tessellation, 109
  - algorithm for, 114–119
  - subdivision for, 111–113
  - watertight, 118–119
- adaptive texture filtering, 434
- addresses
  - floating-point operations for, 514
  - sorting, 516–518
  - translating, 530–534
- aerosols, atmospheric scattering from, 256
- AGP slots, 471
- aliasing. *See also* antialiasing
  - in global illumination, 629–630
  - in lines, 346
  - in Marschner reflectance models, 372
  - in soft-edged shadows, 278
- alpha blending for multiple lines, 355
- alpha channels in refraction simulation, 298–299
- alpha fringes in Motion, 402–405
- alpha gradients in mipmap filtering, 440–441
- alpha tests in mipmap-level measurement, 439, 443
- alpha transparency, 13–15
- alpha values in mipmap-level measurements, 446
- ambient occlusion
  - in deferred shading, 150, 156–157
  - dynamic, 223
    - computations for, 225–226
    - multipass shadow algorithm for, 226–228
    - performance of, 228–230
    - surface elements in, 223–225
  - in dynamic irradiance environment maps, 175
  - in G-buffer optimization, 153
- AMD CodeAnalyst tool, 569
- American options
  - exercising, 721
  - pricing, 725–726
- analytical deformation models, 292–293
- angles in atmospheric scattering, 259
- animation
  - hair. *See* hair animation and rendering
  - in multistreaming, 76
- anisotropy for mipmap levels, 438
- antialiased rasterization, 331–334
  - code for, 339–343
  - downsampling in, 331–336, 341–343
  - filters in, 337–339
  - implementing, 343
  - padding in, 336–337
  - rendering loop in, 340–341
  - tiling and accumulation in, 339
- antialiased shadows, 21
- antialiasing
  - in deferred shading, 158–162
  - in filters, 423–427
  - in GeForce 6 Series GPU, 481
  - in hair generation, 376

antialiasing *(continued)*
in lines. *See* prefiltered lines
for mipmap levels, 438
quasi-optimal, 424–427
approximate bidirectional texture functions, 177–179
rendering in, 181–184
results of, 184–187
setup and acquisition for, 179
shading maps for, 179–181
approximate visibility in occlusion, 105
ARB (Architecture Review Board), 542
`ARB_fragment_program` extension, 396–398, 406, 769
`ARB_occlusion_query` extension, 91, 93
`ARB_pixel_buffer_object` extension, 755
`ARB_vertex_buffer_object` extension, 755
arbitrary meshes in distance-mapping algorithm, 129
architecture drafts. *See* blueprints
Architecture Review Board (ARB), 542
area lights, 231–232
arithmetic intensity
of algorithms, 512
data parallelism for, 494–495
arithmetic operations in kernels, 527
arithmetic units in computation efficiency, 462
arrays, multidimensional, 528–529
1D, 529–530
2D, 530–531
3D, 531–534
higher, 534
artifacts
in distance-mapping algorithm, 127
in Gaussian reconstruction, 432
in lines, 346
from octrees, 605–606, 608–609
from refraction, 296–300
from sampling, 346, 423
from shadows, 272, 274
in soft-edged shadows, 278
in tile-based texture mapping, 198
in water rendering, 290–291
assembler-like languages for Motion, 396
`ATI_draw_buffers` extension, 769
`ATI_texture_float` extension, 769
atmospheric scattering, 253
equations for, 254–255
in-scattering, 257
out-scattering, 256–257
surface-scattering, 257–258
high-dynamic-range rendering in, 265–266
phase function in, 256
Rayleigh scattering vs. Mie scattering, 255–256
real-time, 258–260
shaders in, 260–265
attributes
instancing, 49–50
in Shader Model 3.0 programming, 485
automatic texture-atlas generation, 72

B

B-spline filters
bicubic, 319–320
in cubic convolution, 315–316
in higher-order filtering, 314–315
back buffers for refraction, 296, 300
banded sparse matrices
characteristics of, 706–707
in linear equations, 710–711
banding artifacts, 272
bandlimiting signals, 347–349
bandwidth
in antialiasing, 423
in arithmetic intensity, 494
of computer system components, 472
in deferred shading, 155
in DRAM performance, 458
for lookups, 565
trends in, 459
barrel distortion, 651–653
basis functions in spherical harmonic convolution, 171
batches and batching
in geometry instancing, 50, 52–53
API for, 61–65
dynamic, 56–57
static, 54–56, 66
in tessellation, 122
Bayesian-matting, 384

benchmarking
in FFT, 775–776
in optimizations, 568
bending light. *See* refraction simulation
bent normals in ambient occlusion computations, 225, 227
Bezier curves for hair generation, 220, 364–365
BGRA texels in LBM, 751
biasing problems in deferred shading, 155
bicubic filtering
B-spline, 319–320
reconstruction, 323
bidirectional reflectance distribution functions (BRDFs)
for material properties, 178
in material rendering, 181–183
for Phenomena, 202
for shading maps, 180
bidirectional texture functions (BTFs). *See* approximate bidirectional texture functions
bilinear filtering
for depth textures, 277
in GeForce 6 Series GPU, 482
in Motion, 400–405
packing heights for, 288–290
reconstruction, 323
for texture tiles, 192
bilinear interpolation, 404
bilinear texture
in cubic convolution, 317
in water rendering, 289
billboards for grass layer, 11–13
binding
domains, 504
vertex buffers to streams, 81
binomial lattice models, 725–727
bitonic sort, 579, 742
algorithm for, 739
grouping keys in, 740–741
implementing, 743–744
Black, Fischer, 721
Black & White 2 game, 47–48
Black-Scholes model, 721–724
blending
in blueprints, 242
in compositing lines, 355
in GeForce 6 Series GPU, 477, 480, 483
for geometry clipmaps, 28–29
block diagram of GeForce 6 Series GPU architecture
graphics operations, 473–477
non-graphics operations, 478–480
blocking strategy for static sparse structures, 537
bloom targets in deferred shading, 162
blueprints
composing, 241–242
depth masking for, 242–244
depth peeling for, 238–241
edge extraction for, 241
rendering, 235–236
for architecture, 244
depth sprites, 237–238
edge enhancement in, 236–237
intermediate results, 236
blurring
in deferred shading, 158–159, 162
hard-edged shadows, 271–274
in Motion, 395, 400–402
for soft-edged shadows. *See* soft-edged shadows
in virtual botany, 22
Boolean conditions with z-culls, 552
botany. *See* virtual botany
bottlenecks
in deferred shading, 147
memory, 524
in optimization, 569
boundaries
complex. *See* complex boundaries
for geometry clipmaps, 28
for prefiltered lines, 357
boundary distributions, 758
bounding boxes, 581–583
in hierarchical stop-and-wait function, 96
for painting on meshes, 603
bounding polygons, 679–686
bounding triangles, 684–685
branching
in deferred shading, 150
in flow control, 548–550, 553–554
in GeForce 6 Series GPU, 490–491
inner-loop, 566
instructions for, 553–554

branching *(continued)*
- in pipelines, 275, 549–550
- pixel shader. *See* pixel shaders
- in Shader Model 3.0 programming, 485, 487–488
- in water rendering, 290

BRDFs (bidirectional reflectance distribution functions)
- for material properties, 178
- in material rendering, 181–183
- for Phenomena, 202
- for shading maps, 180

brightness, lookup tables for, 384–385

Brook language, 579–580

BSP trees in hierarchical stop-and-wait function, 95

BTFs (bidirectional texture functions). *See* approximate bidirectional texture functions

buckets
- in antialiased rasterization, 331–333, 339
- in sparse data structures, 538–539

buffers
- in antialiased rasterization, 333, 336, 339
- binding to streams, 81
- in computer vision, 655–656
- in deferred shading, 143, 145–146
- in GeForce 6 Series GPU, 474, 479–480, 482–483
- for geometry clipmaps, 30–31, 33–34
- in geometry instancing, 58, 62
- for grass layer, 12–13
- for hardware occlusion queries, 92
- in multistreaming, 85–88
- optimizing, 151–153
- pixel, 541–543
- in protein structure prediction, 698
- for refraction, 296, 300
- segment buffering. *See* segment buffering
- for stream management, 504
- for visible points data, 628
- for voxelization, 755

bump mapping, 206, 216–217

bump texture in glass simulation, 303

butterfly lookups, 770, 773

C

caches
- challenges in, 468
- in communication efficiency, 462–463
- in CPU performance, 464
- in GeForce 6 Series GPU, 475
- GPU vs. CPU, 510–511

calibration data in mipmap-level measurements, 444–446

call options, 720

cameras
- in material acquisition, 179
- variables for, 212
- for virtual botany, 9

candidate pixels, 479

Canny edge-detection algorithm, 653–655

capability, trends in, 458

Cartesian grid simulation, 495–496

Catmull-Clark subdivision surfaces, 109–111

cells in indirection pools, 598

central differencing schemes
- for gradient information, 324
- in partial differential equations, 714–715

centroids
- computing, 657
- sampling, 485

Cg language for kernels, 503–504

Cg shaders
- for lookup tables, 386–388
- for prefiltered lines, 354–355

Cg structures and Phenomena, 205

Cg vertex program
- bump mapping in, 216–217
- global state variables in, 211–212
- hair rendering in, 220
- parameters in, 205–206
- shaders in
  - environment and volume, 217–218
  - example, 208–210
  - fragment, 207
  - general interfaces for, 207–208
  - light, 211, 213–214
  - returning structures, 218–219
  - texture, 215–216

CgFX for Perlin noise, 411

chaining operations, 465–466

children in octrees, 597–598

chip generations, trends in, 458

choppiness in water rendering, 293

clamping with lookup tables, 390

`clip()` function, 551

clip planes for voxelization, 754

clip space in conservative rasterization, 681
clipmaps, geometry. *See* geometry clipmaps
clipping, 481
clocks
 in GeForce 6 Series GPU, 488
 speed trends in, 458
cloth
 material, 184
 simulating, 369
cloud simulation, 496
clustering of final gathering rays, 621–623
code books for compression, 668
coherence for dynamic branching, 276
coherent hierarchical culling
 algorithm for, 99–100
 guessing in, 97–98
 implementation details for, 100–103
 pull up in, 98–99
 queries in, 104
 stalls in, 103
 traversals in, 104–105
coissue in Shader Model 3.0 programming, 485–486
collisions
 detecting, 579–583
 in hair generation, 367–368
color
 in blueprints, 239, 242
 extrapolating, 609–610
 in geometry clipmaps, 38–39
 in glass simulation, 303
 in grass layer, 15
 HSV color space, 656
 layer maps for, 239
 in multistreaming, 80
 in octree texture conversions, 609–610
 in sketchy rendering, 245, 247–248
color banding in soft-edged shadows, 277
color-space ceiling, 390
color transformations
 lookup tables for. *See* lookup tables (LUTs)
 in Motion, 394–395
commands in GeForce 6 Series GPU, 474
communication
 in arithmetic intensity, 494
 challenges in, 468
 efficiency in, 462–463
 in stream programming model, 466–467, 496
 trends in, 459–460
compaction, filtering through, 574–579
complex boundaries, 753–754
 obstacle, 757–759
 outflow, 756–757
 periodic, 756
 results, 760–761
 visualization in, 759–760
 voxelization in, 754–756
composing blueprints, 241–242
compositing multiple lines, 355–356
compression
 in communication efficiency, 463
 for geometry clipmaps, 29, 31, 40
 keys for, 668–669
 texture, 189
computation
 in arithmetic intensity, 494
 challenges in, 468
 efficiency in, 461–462
 in fast algorithms, 511–512
 in global illumination, 641–643
 for noise, 410–411
 in stream programming model, 464, 466–467
 trends in, 458–460
computational domains, 504
computational frequency, 561–563
 inner-loop branching, 566
 lookup table precomputation, 564–565
 loop-invariant precomputation, 563–564
 in performance, 541
 swizzle operator, 566–568
computational resources, 497–500
compute-and-sum operations, 660
computed tomography (CT), 765
computer-mediated reality, 651
computer vision, 649–650
 Canny edge-detection algorithm for, 653–655
 feature vectors for, 661–663
 fragment program sequences for, 651–655
 hand tracking in, 656–658
 image panorama equation for, 658–660
 implementation framework for, 650–651
 parallel computer vision processing, 664
 radial distortion in, 651–653
 summation operations in, 655–658
computing, 509
 data parallel, 557–561

computing *(continued)*
  fast algorithms for, 509–513
  floating-point operations, 513–515
  high-performance, 461
    communication efficiency in, 462–463
    computation efficiency in, 461–462
    CPUs, 463–464
  scatter operations, 515–518
condition codes for branching, 549
conditional execution in Motion, 395
conjugate gradient solver, 713–714
conservative rasterization, 677–678
  algorithms for, 679–681
    clip space, 681
    first attempt, 681–683
    second attempt, 683–686
  conservative depth in, 687–689
  implementation, 681–683
  problem definition in, 678–679
  results and conclusions, 689–690
  robustness issues in, 686–687
constant parameters and values
  computational frequency of, 541
  in GPGPU framework, 505
constraints in hair generation, 366–367
contexts
  in geometry instancing, 50–51
  for pbuffers, 542
continuity in deferred shading, 154–155
control
  in computation efficiency, 462
  flow. *See* flow control
control hairs, 361, 364–365
converters for shaders, 219
converting
  addresses for packed representation, 533
  octree textures into standard 2D textures, 607–610
  polygonal data to surface elements, 223–225
  scattering to gather, 515–516
convex hulls, 681
convolution
  in dynamic irradiance environment maps, 173–175
  recursive cubic, 315–320
convolution filters, 395, 419–422
convolution sums, 325
convolution theorem, 171
Cooley-Tukey algorithm, 767
coordinate-space conversions, 211
coordinates
  texture. *See* textures
  transforming, 658–659
copy-to-vertex-buffer operation, 525
core technology trends, 458
CPU-based occlusion culling, 148
CPU data in GPGPU framework, 505
CPU fallback, 396–397
CPU stalls, 96
CPUs
  in deferred shading, 144
  GPU analogies for, 500–503
  performance of, 463–464
  in tessellation algorithm, 114
Crank-Nicholson scheme, 716–717
CT (computed tomography), 765
cube maps
  in deferred shading, 154
  in dynamic irradiance environment maps, 175
  in glass simulation, 303
cubic convolution, 315–320
cubic filters, 314–317
culling
  coherent. *See* coherent hierarchical culling
  in GeForce 6 Series GPU, 481–482, 490
  occlusion, 93–94, 157
  view frustum, 35
  z-culls, 550–553
curvature information in derivative reconstruction, 326
curved-surface subdivisions, 109

D

D2Q9 model, 748, 750, 760–761
D3Q19 model, 749, 751–752, 760
data caches. *See* caches
data-dependent looping, 554–555
data-driven sorting algorithms, 734
data filtering. *See* filters and filtering
data flow
  in hair generation, 364
  in stream programming model, 465
data-independent sorting algorithms, 734
data packing for vectorization, 699–700

data parallelism, 522–523
  for arithmetic intensity, 494–495
  in computation efficiency, 462
  for grid simulation, 495–496
  importance of, 493–494
  in optimization, 557–561
  for stream communication, 496
data storage formats, 488–489
data structures, 534–535
  for geometry clipmaps, 31
  sparse, 535–536
    dynamic, 537–540
    static, 536–537
  streams of, 534–535
data textures, indexing, 698–699
data types, 499
datapaths in computation efficiency, 462
deblurring images, shock filtering for, 430–432
debugging in Motion, 406–407
decimation-in-time butterfly algorithm, 767–768, 773
decoding costs in G-buffer optimization, 152–153
deferred filtering, 667–668
  algorithm, 669–672
  operation of, 672
  purpose of, 668–669
deferred shading, 143–145
  ambient occlusion in, 150, 156–157
  antialiasing in, 158–162
  efficient tone mapping in, 162
  elevation maps in, 163
  materials in, 146, 157–158
  myths about, 145–146
  optimizations for, 147–154
  quality in, 154–158
  real-time global illumination in, 163–164
  surface-light interaction in, 157–158
  transparency in, 162
  virtual positions in, 155–156
deforming implicit surfaces, 539
degree of uncertainty in sketchy rendering, 246
dense matrices, 706
dense structures, 535
dependencies in coherent hierarchical culling, 103
dependent textures, 528
  lookups, 386
  reads, 540–541
depth
  in conservative rasterization, 687–689
  in G-buffer optimization, 151–152
  in GeForce 6 Series GPU, 480
  in sketchy rendering, 247
depth buffers
  in GeForce 6 Series GPU, 479
  for hardware occlusion queries, 92
depth-compare/blending data write engine, 478
depth-D grids for octrees, 599
depth-first approaches in hierarchical traversals, 104
depth layer maps, 239
depth layers
  for blueprints, 239
  in final gathering, 623–624
depth maps in hair generation, 379
depth masking for blueprints, 242–244
depth-of-field effects in global illumination, 628
depth peeling
  in blueprints, 238–241
  in deferred shading, 162
  in global illumination, 624, 626
  in voxelization, 754–755
depth sprite rendering
  implementing, 237–238
  in sketchy rendering, 247
depth tests
  for blueprints, 240–241
  in GeForce 6 Series GPU, 477
derivatives, 719–720
  in mipmap-level measurements, 447
  in third-order texture filtering, 324–327
  in vertex program, 206
detecting
  collisions, 579–583
  infinity, 574
device contexts for pbuffers, 542
DFT (Discrete Fourier Transform), 767
diffuse color in multistreaming, 80
diffuse irradiance environment maps, 168–169
diffuse surfaces, indirect lighting off, 231
diffuse texture in glass simulation, 303
diffusion in reaction-diffusion model, 505–508
digital image resampling, 422–423
  antialiasing in, 423–427
  image reconstruction in, 427–430
dilation in conservative rasterization, 679

dimensions
in atmospheric scattering, 260–261
of lookup tables, 382
in octrees, 597
direct illumination, 629
direct memory access (DMA) in multistreaming, 78–79
directional illumination, light shaders for, 213
directional lights, deferred shading for, 146
DirectX 9
for geometry instancing, 61–62
multistreaming with, 78–81
discard keyword, 551, 554
discontinuity edges for prefiltered lines, 357
discounted option values, 720
Discrete Fourier Transform (DFT), 767
displaced values, searching for, 577–578
displacement mapping, 109–110
with distance functions. *See* distance functions
for subdivision surfaces, 119–121
in water rendering, 283–284, 293
dissolves for alpha transparency, 13–15
distance-bound smoothing, 697
distance functions, 123–125
distance map computations, 129
distance-mapping algorithm, 126–129
previous work, 125–126
results with, 132–134
shaders for, 130–132
distances in scaling, 125
distortion in computer vision, 651–653
distributions, boundary, 758
division by zero, 399
DMA (direct memory access) in multistreaming, 78–79
domains
binding, 504
for lookup tables, 385
download in fast algorithms, 512–513
downsampling in antialiased rasterization, 331–336, 341–343
DRAM (dynamic random-access memory)
in GeForce 6 Series GPU, 477
trends in, 458
driver optimizations, 438
dsx instruction, 447
dsy instruction, 447
dual issue in Shader Model 3.0 programming, 486
dual-paraboloid shadow masks, 154
dust, atmospheric scattering from, 256
DYANA package, 695
dynamic ambient occlusion, 223
computations for, 225–226
multipass shadow algorithm for, 226–228
performance of, 228–230
surface elements in, 223–225
dynamic batching, 56–57
dynamic branching
in deferred shading, 150
in fragment programs, 275
dynamic displacement mapping, 293
dynamic flow control, 483, 485
dynamic irradiance environment maps, 167–169
further work for, 175
GPU mapping in, 172–175
spherical harmonic convolution in, 170–172
dynamic random-access memory (DRAM)
in GeForce 6 Series GPU, 477
trends in, 458
dynamic scenes in occlusion culling, 93
dynamic sparse structures, 537–540
dynamic updates in protein structure prediction, 698
dynamics/collision computations in hair generation, 361

E

early culling, in GeForce 6 Series GPU, 479, 481
edge detection
for computer vision, 653–655
in deferred shading, 159–161
in Motion, 395
edge filters for computer vision, 653
edge maps, 237
edges
in conservative rasterization, 680
enhancement of, 236–237
extracting, 241
with hemispherical projection, 641
for prefiltered lines, 357
in sketchy rendering, 245, 248
efficiency
in communication, 462–463, 466
in computation, 461–462, 466

efficiency *(continued)*
  in soft-edged shadows, 274–277
  in stream programming model, 466
efficient tone mapping, 162
elevation angles in material acquisition, 179
elevation data for geometry clipmaps, 40
elevation maps
  in deferred shading, 163
  for geometry clipmaps, 30
emitters in ambient occlusion computations, 225–226
emitting modified draw calls, 443–444
emulators, 397
encoding indices for octrees, 602
environment maps
  dynamic irradiance. *See* dynamic irradiance environment maps
  in glass simulation, 303
environment shaders
  in Cg vertex program, 217–218
  and Phenomena, 202
equations
  for image panoramas, 658–660
  linear. *See* linear equations
erosion in conservative rasterization, 679
estimation for soft-edged shadows, 276–278
European options
  Black-Scholes model, 721–724
  exercising, 721
  lattice models, 726–730
exercise restrictions on options, 721
expansion operations in stream programming model, 464
expiration date of options, 720
exponential functions, 484
exponents in floating-point representation, 513–514
exterior regions in sketchy rendering, 246
extracting edges, 241
extraordinary points
  for surfaces, 110
  in tessellation performance, 122
extrapolation in octree texture conversions, 609–610
eye space in G-buffer optimization, 152–153

F

false-colored mipmaps, 437–439
Far Cry game, 295, 300
fast algorithms, 509
  computation emphasis in, 511–512
  download and readback in, 512–513
  memory performance in, 510–511
fast filtering schemes, 326
Fast Fourier Transform. *See* FFT (Fast Fourier Transform)
fast sorting, 735
fat-format encoding, 151
fat lines, drawing, 355
feature vectors for computer vision, 661–663
fetch instructions, latency of, 511
fetching data in GeForce 6 Series GPU, 475–476
FFT (Fast Fourier Transform), 765–767
  algorithm, 767–768
  benchmarking in, 775–776
  fragment processors for, 770–772
  implementing, 768–770
  load balancing in, 775
  in medical imaging, 776–780
  in ultrasonic imaging, 780–783
  vertex processors, rasterizers, and fragment processors for, 772–775
  in water rendering, 283
filters and filtering
  antialiasing in, 337–339, 423–427
  bicubic
    B-spline, 319–320
    reconstruction, 323
  bilinear. *See* bilinear filtering
  compaction for, 574–579
  for computer vision, 650, 653
  convolution, 395, 419–422
  cubic, 314–317
  deferred. *See* deferred filtering
  in deferred shading, 150, 155, 158–159
  digital image resampling for, 422–423
    antialiasing in, 423–427
    image reconstruction in, 427–430
  with distance functions, 132
  in GeForce 6 Series GPU, 477–478, 482, 490
  image samples for, 418–419
  implementing, 417–418, 433
  for lines, 351
  motion blur removal, 434
  performance of, 579
  shock, 430–432
  in stream programming model, 464

- filters and filtering *(continued)*
  - for subdivision surfaces, 583–587
  - for textures, 192
    - adaptive, 434
    - in loading, 490
    - for mipmap levels, 438
    - third-order. *See* third-order texture filtering
    - types of, 314–315
  - time warping, 433–434
  - tricubic, 317, 324
- final gathering in global illumination, 618–624
- finalizing applications, 503
- fins, hair generation for, 368–370
- first-order linear filtering, 314–315
- first pass in global illumination, 628
- fixed-function features, 481–483
- fixed-function units, limitations of, 467
- fixed-precision math, 477
- flat ambient lighting, 175
- flatness tests
  - in displacement mapping, 120
  - in tessellation algorithm, 114–115
- flickering in global illumination, 630–632
- floating-point formats, 499
- floating-point infinity, 574
- floating-point operations
  - in GeForce 6 Series GPU, 477
  - working with, 513–515
- flow control, 547
  - branching in, 548–550, 553–554
  - challenges in, 547–549
  - data-dependent looping, 554–555
  - predication for, 549
  - in Shader Model 3.0 programming, 483, 485
  - z-culls, 550–553
- flow simulation
  - with complex boundaries, 753–754
    - obstacle, 757–759
    - outflow, 756–757
    - periodic, 756
    - results, 760–761
    - visualization in, 759–760
    - voxelization in, 754–756
  - with Lattice Boltzmann Method, 747–749
    - algorithm overview, 749–751
    - packing in, 751–752
    - streaming in, 752–753
- Floyd-Warshall algorithm, 697
- fluid dynamics, 283
- foam generation, 293
- fog
  - in deferred shading, 146
  - and Phenomena, 203
  - volume shaders for, 218
- fog blending, 477
- fog units, 477
- forecasts in soft-edged shadows, 276–277
- form factors
  - in global illumination, 641–643
  - for radiosity, 636
- forward rendering in deferred shading, 162
- forward shading engines, 144
- four-point subdivision curve interpolants, 40
- four-sample bilinear percentage-closer filtering, 277
- fp16 surfaces, 483
- `frac` instruction for octrees, 602
- fractal details for geometry clipmaps, 42
- fractional filter radii, 336–337
- fractional shadow values, 376
- `fragment()` function, 196
- fragment processors
  - characteristics of, 497–498
  - for FFT, 770–775
  - in GeForce 6 Series GPU, 475–478, 490–491
  - in Shader Model 3.0 programming, 484–488
- fragment programs
  - for computer vision, 651–655
  - CPU-GPU analogies, 500
  - in options pricing, 728–729
  - in stream programming, 467
- fragment shaders
  - in atmospheric scattering, 264–265
  - for distance functions, 130–131
  - main entry point for, 207
- fragment streams
  - characteristics of, 526
  - computational frequency of, 541
- fragments
  - in GeForce 6 Series GPU, 478–480
  - vertices for, 497
  - in voxelization, 754
- frame-buffer blending, 355
- frame-buffer memory, 474
- frame-buffer streams, 526
- frame rate for geometry clipmaps, 43
- free components in G-buffer optimization, 152–153

free four-sample bilinear percentage-closer filtering, 277
free tiles, 539
Fresnel terms in refraction simulation, 300–303
fringes, alpha, 402–405
front-to-back order in hierarchical traversals, 104
full-function attributes, 485
full matrices, 706
full matrix-vector products, 710–711
full-scene glow, 23–24
full vectors, 706
functional block diagrams of GeForce 6 Series GPU architecture
  graphics operations, 473–477
  non-graphics operations, 478–480
functionality, challenges in, 468–469
future technology trends, 468

G

G-buffers
  in deferred shading, 143, 145–146
  optimizing, 151–153
G streams in multistreaming, 76–78, 84, 89
gamma tables, 382
gathering
  in global illumination
    final, 618–619
    rasterization in, 621–624
    two-pass methods in, 617–621
  scatter converted to, 515–516
  in stream communication, 496
Gaussian blur
  in Motion, 395
  in virtual botany, 22
Gaussian filters
  in antialiased rasterization, 334–335
  for computer vision, 653
  for prefiltered lines, 348, 351, 356, 358
Gaussian noise for geometry clipmaps, 42
Gaussian reconstruction, 428–429, 432
Gaussian subpixels, 420–421
GeForce 6 Series GPU
  architecture, 471
    in computer systems, 471–473
    fixed-function features in, 481–483
    graphics operations, 473–477
    non-graphics operations, 478–480
  data storage formats for, 488–489
  performance of, 489, 491
  Shader Model 3.0 programming model in, 483–488
GeForce FX performance trends, 459
Gelato film renderer, 331–332
general-purpose applications, CPUs for, 463
general-purpose programming on GPUs, 293. *See also* GPGPU (general-purpose programming on GPUs)
general shader interfaces, 207–208
geometric term for radiosity form factor, 636
geometry
  in hair generation, 361
  in hardware occlusion query optimizations, 105
  in multistreaming, 76, 85, 89
  in sketchy rendering, 249–250
geometry clipmaps, 27–29
  active levels for, 32
  buffers for, 30–31, 33–34
  data structures for, 31
  `DrawIndexedPrimitive` calls for, 50, 55–56, 63
  `DrawPrimitive` calls for, 35
  GPU implementation for, 30–32
  normal maps for, 42, 44
  pixel shaders for, 28, 38–39, 41
  rendering, 32–39, 43
  residuals for, 42
  results and discussion, 43
  size of, 31–32
  summary, 43–44
  updates for, 39–42
  upsampling for, 40–41
  view frustum culling for, 35
geometry instancing, 47–48
  attributes for, 49–50
  batching in, 50, 52–53
    API for, 61–65
    dynamic, 56–57
    static, 54–56, 66
  in GeForce 6 Series GPU, 481
  geometry packets in, 49
  implementing, 53–54
  purpose of, 48
  render and texture contexts in, 50–51
  vertex constants instancing in, 57–61

geometry rasterization, 501
GL_ extensions, 769–770
glass simulation, 303–305
global communication, 466
global illumination, 615–616
  aliasing in, 629–630
  depth peeling in, 626
  first pass in, 628
  flickering and popping in, 630–632
  gathering in
    implementation details, 625–627
    rasterization in, 621–624
    two-pass methods in, 617–621
  initialization in, 625–626
  performance of, 627, 645
  progressive refinement radiosity for, 635–637
    adaptive subdivision in, 643–645
    form factor computation in, 641–643
    foundations of, 636–638
    GPU implementation, 638–639
    next shooters in, 643
    progressive refinement in, 637–638
    quadtree subdivision in, 644–645
    visibility using hemispherical projection, 639–641
  sampling in, 627
  second pass in, 629
  visible points data in, 628
global radii for hair rendering, 220
global ray direction in final gathering rays, 621
global shadowing terms, 175
global state variables, 211–212
globally unique identifiers (GUIDs), 81
gloss masks, 153
glossy specular reflections, 148
Gothic III, 75, 81, 88
GPGPU (general-purpose programming on GPUs), 293
  computational domains and ranges in, 504
  constant parameters in, 505
  CPU data in, 505
  example, 505–508
  initializing and finalizing applications, 503
  kernel invocation in, 505
  kernel specifications in, 503–504
  parallel reductions in, 505
  stream management in, 504
GPUs
  CPU analogies for, 500–503
  in dynamic irradiance environment maps, 172–175
  functionality assumption by CPU, 470
  GeForce. *See* GeForce 6 Series GPU
  pixel-processing power of, 125
  sorting. *See* sorting
  in tessellation performance, 122
gradients
  conjugate gradient solver, 713–714
  in mipmap filtering, 440–441
  for Perlin noise, 411–413
  in volume rendering, 324
graphics clocks, 488
graphics connectors, 471
graphics pipelines, 465, 467
grass layer, 11–13
  alpha transparency simulations for, 13–15
  lighting for, 15–16
  variation in, 15
  wind for, 17
greater-equal comparisons in sorts, 735, 740
Grey-Scott model, 505–508
grids
  data parallelism for, 495–496
  for geometry clipmaps, 28, 31, 33–34
  for virtual botany, 17–18
ground cover, 17–18
grouping keys in bitonic merge sorts, 740–741
guard-band spaces in water rendering, 290
GUIDs (globally unique identifiers), 81
gyromagnetic ratio, 777

H

hair animation and rendering, 361
  collisions in, 367–368
  constraints in, 366–367
  controlling, 362–363
  data flow in, 364
  for fins, 368–370
  interpolation in, 364–366
  layout and growth of, 362
  reflectance models for, 369–375
  rendering, 220
  shading in, 369–378

hair animation and rendering *(continued)*
  tessellation in, 364–365
  volumetric shadows in, 375–378
hand tracking in computer vision, 656–658
hard-edged shadows, blurring, 271–274
hardware
  in CPU performance, 463
  in deferred shading, 155
  in optimization, 152–153, 568
hardware occlusion queries, 91–92
  coherent hierarchical culling in. *See* coherent hierarchical culling
  hierarchical stop-and-wait function, 94–97
  occlusion culling in, 93–94
  optimizations in, 105–106
  scenes for, 92
harmonics in water surface models, 284
hashing
  for Perlin noise, 411
  for texture tile mapping, 195
haze
  atmospheric scattering from, 256
  refraction for, 295
HDR (high-dynamic-range) data
  in atmospheric scattering, 265–266
  blending, 483
  lookup tables for, 390–391
head, MRI for, 779–780
Heapsort algorithm, 734
heat haze, refraction for, 295
height maps for water
  in rendering, 286–287
  surface models, 284–285
hemispherical clustering of final gathering rays, 621
hemispherical projection for visibility, 639–641
Hermite polynomials, 412
Hero's Journey, 7–8, 14
hidden objects in occlusion culling, 93
hierarchical culling. *See* coherent hierarchical culling
hierarchical occlusion-culling systems, 148
  benefits of, 95
  naive algorithm for, 94–95
  stalls in, 96
hierarchy, memory, 524
high-dynamic-range (HDR) data
  in atmospheric scattering, 265–266
  blending, 483
  lookup tables for, 390–391
high-frequency components, shock filtering for, 430–432
high-frequency limiting for prefiltered lines, 347
high-latency operations, 467
High-Level Shader Language (HLSL) vertex program, 37–38, 41
high-performance computing, 461
  communication efficiency in, 462–463
  computation efficiency in, 461–462
  CPUs, 463–464
high-precision storage, 405–406
higher-dimensional arrays, 534
higher-order filtering. *See* third-order texture filtering
highlights, lookup tables for, 384–385
HLSL (High-Level Shader Language) vertex program, 37–38, 41
holders of options, 720
Horner's rule, 723
HSV color space, 656
hue in HSV color space, 656
human head, MRI for, 779–780

I

I streams in multistreaming, 76–78, 84
`IDirect3DQuery9` interface, 91
illumination, global. *See* global illumination
image-based lighting, 629
image buffers in computer vision, 655–656
image panoramas, 658–660
image reconstruction, 427–430
images in Motion, 394
implicit surfaces, 538–539
in-scattering equation, 257
in-slice periodic boundary faces, 756
independent neighborhood operations, 395
index buffers
  for geometry clipmaps, 31, 33–34
  for grass layer, 12
index data and indices
  for attributes in Shader Model 3.0 programming, 485
  in geometry packets, 49
  in multistreaming, 88–89
  for octrees, 602

index data and indices *(continued)*
  for Perlin noise, 411
  for textures, 698–699
index of refraction. *See* refraction simulation
index texture maps, 195
indirect lighting, 223
  area lights in, 231–232
  dynamic ambient occlusion for, 225–230
  in global illumination, 629
  surface elements in, 223–225
indirect writes. *See* scatter and scattering
indirection grids for octrees, 598
indirection levels in structures, 536
indirection pools for octrees, 598–600
infinity detection, 574
inner-loop branching, 566
input scrambler stage for FFT, 767
input tiles in texture mapping, 197
inputs in Motion, 394
instances in segment buffering, 72
instancing
  in GeForce 6 Series GPU, 481
  geometry. *See* geometry instancing
  in multistreaming, 76
  in Shader Model 3.0 programming, 483
instruction counts in Shader Model 3.0 programming, 483–484
instruction parallelism
  in computation efficiency, 462
  in CPU performance, 463
  in optimization, 558–559
instruction pipelines, flow control in, 548
integer data types, 499
Intel VTune tool, 569
interface classes for linear equations, 712–713
interface parameters and Phenomena, 204
interior nodes
  in coherent hierarchical culling, 102
  in hierarchical stop-and-wait function, 95–96
interior regions in sketchy rendering, 246
intermediate results
  in GeForce 6 Series GPU, 491
  in stream programming model, 466
internal nodes of octrees, 597–598
internal precision in Shader Model 3.0 programming, 485
International Technology Roadmap for Semiconductors (ITRS), 457
interpolation
  cost of, 563
  for geometry clipmaps, 36
  in hair generation, 364–366
  for lookup tables, 384–386
  in Motion, 404
  in painting on meshes, 605–606
invariant kernels, 1D textures for, 422
inverse Fourier transform, 766
inverse spherical harmonic transforms, 172
invoking kernels, 505
irradiance environment maps, dynamic. *See* dynamic irradiance environment maps
isosurfaces, 538
ITRS (International Technology Roadmap for Semiconductors), 457

J

jaggies, 345
jeans, rendering of, 184
jittered samples for soft-edged shadows, 274–278

K

k-d trees
  in hierarchical stop-and-wait function, 95
  in segment buffering, 71
kernels
  CPU-GPU analogies, 500
  invariant, 422
  invoking, 505
  memory access by, 527–528
  specifying, 503–504
  in stream programming, 464–467, 522
keys
  in compression, 668–669
  in sorts, 740–741, 743

L

languages in Motion, 396
Laplacian operator, 506
large kernels, 422
large textures, 191

largest numbers allowed, 514
Larmor frequency, 777
latency
  in CPU performance, 464
  in DRAM performance, 458
  in fetch instructions, 511
  in GeForce 6 Series GPU, 476
  in mipmap-level measurements, 444
  trends in, 459
Lattice Boltzmann Method (LBM), 747–749
  algorithm overview, 749–751
  packing in, 751–752
  streaming in, 752–753
lattices, 725
  binomial model, 725–727
  for European options, 726–730
  in LBM model. *See* Lattice Boltzmann Method (LBM)
  for lookup tables, 389–391
layering in deferred shading, 163
LBM (Lattice Boltzmann Method), 747–749
  algorithm overview, 749–751
  packing in, 751–752
  streaming in, 752–753
leaf cluster images, 19
leakage, shadow, 278
leather material, 184
leaves and leaf nodes
  in hierarchical stop-and-wait function, 96
  in octrees, 597–598
lens shaders, 202
less-than comparisons in sorts, 734–735, 740
level-of-detail (LOD)
  in geometry instancing, 55
  for tree and shrub layers, 18–19
  in water rendering, 287, 289–290
levels of indirection in structures, 536
levels of mipmapping in painting on meshes, 606–607
Leviathan engine, mipmap levels with, 442–444
light
  bending. *See* refraction simulation
  in Marschner reflectance models, 371–372
  scattering. *See* atmospheric scattering
light maps
  in deferred shading, 157
  in multistreaming, 86
light shaders
  in Cg vertex program, 211, 213–214
  and Phenomena, 203
  variables for, 212
lighting and light sources
  in Cg vertex program, 220
  in deferred shading, 146–150, 154
  environment maps for. *See* dynamic irradiance environment maps
  global. *See* global illumination
  for grass layer, 15–16
  indirect. *See* indirect lighting
  in material acquisition and rendering, 179, 181
  for soft-edged shadows, 278
limit positions in subdivision tessellation algorithm, 118
limit surfaces, 110
linear equations, 703–704
  operations in, 708
    conjugate gradient solver, 713–714
    matrix-vector products, 710–712
    vector arithmetic, 709
    vector reduce, 709–710
  partial differential equations, 714–717
  representation for, 704
    matrices, 706–708
    single float, 704
    vectors, 704–705
linear filtering, 314–315
linear interpolation
  in lookup tables, 385
  in painting on meshes, 605–606
linear mipmap filtering, 440
linear searches in relief mapping, 126
linear transformations
  in antialiasing, 425
  in recursive cubic convolution, 317
lines
  in GeForce 6 Series GPU, 475
  prefiltered. *See* prefiltered lines
liquid flowing simulation, 611–612
load balancing
  in FFT, 775
  in optimization, 568–570
loading
  fragment processor, 770–772
  kernel programs, 504

loading *(continued)*
vertex processors, rasterizers, and fragment processors, 772–775
local frame-buffer memory, 474
local perturbations in water rendering, 292–293
LOD pools in painting on meshes, 607
LOD (level-of-detail) schemes
in geometry instancing, 55
for tree and shrub layers, 18–19
in water rendering, 287, 289–290
lookup for octrees, 599–601
lookup tables (LUTs)
basics, 381–382
Cg shaders for, 386–388
for color transformation, 381
high-dynamic-range imagery with, 390–391
interpolation for, 384–386
mapping, 386
in Marschner reflectance models, 373–375
one-dimensional, 382–383
optimization for, 387–388
for precomputation, 564–565
for prefiltered lines, 353–355
system integration for, 388–390
three-dimensional, 383–385
lookup textures, 317, 327–328
loop invariants, precomputation of, 563–564
loops
data-dependent, 554–555
for displaced values searches, 578
in flow control, 548
for rasterization, 562
low-pass filters
in deferred shading, 158–159
for prefiltered lines, 347–348, 351
in Shannon-Nyquist signal reconstruction function, 428
low-polygonal geometry in deferred shading, 157
LUTs. *See* lookup tables (LUTs)

M

magnetic resonance imaging (MRI), 765–766
for human head, 779–780
for mouse heart, 778–779
operation of, 776–778
magnification filtering, 328
magnification in mipmap-level measurements, 445
main root shaders, 204
mantissas in floating-point representation, 513–514
many-to-one mappings for feature vectors, 661
maps and mapping
bump, 206, 216–217
in deferred shading, 154, 157, 162–163
depth, 379
displacement, 109–110
per-pixel with distance functions. *See* distance functions
for subdivision surfaces, 119–121
in water rendering, 283–284, 293
dynamic irradiance. *See* dynamic irradiance environment maps
for feature vectors, 661
for geometry clipmaps, 30
in glass simulation, 303
height, 284–287
lookup tables, 386
in multistreaming, 86
shadow. *See* shadow maps
in stream programming, 464–465, 467
texture
lookup tables for, 564–565
in multistreaming, 76
tile-based. *See* tile-based texture mapping
tone, 162
Marschner reflectance models, 369–375
masks and masking
in deferred shading, 150, 154–155
depth, 242–244
in G-buffer optimization, 153
for geometry clipmaps, 40
in refraction simulation, 297–300
material shaders
in Cg vertex program, 211
and Phenomena, 203
materials
acquisition of, 179–181
in deferred shading, 146, 157–158
discussion, 186–187
in G-buffer optimization, 153
in realistic images, 177
rendering process, 181–184
rendering results, 184–186
variables for, 212

matrices in linear equations, 706
  full, 706
  sparse, 706–708
matrix-vector products, 710–712
medical image reconstruction
  MRI, 765–766
    for human head, 779–780
    for mouse heart, 778–779
    operation of, 776–778
  ultrasonic imaging, 780–783
memory
  in CPU performance, 463–464
  in deferred shading, 155
  in GeForce 6 Series GPU, 474, 477
  hierarchy of, 524
  kernel access to, 527–528
  for lookups, 565
  in mipmap-level measurements, 448
  performance of, 510–511
  stream types, 525–526
  trends in, 458
memory clocks, 488
memory footprints
  in geometry instancing, 55
  texture, 322
memory-free terrain synthesis, 44
memory interface for GPUs, 472
mental images Phenomena renderer
  implementing. *See* Cg vertex program
  and shaders, 202–205
meshes
  in distance-mapping algorithm, 129
  in multistreaming, 76, 82–83
  painting on. *See* painting on meshes
  in refraction simulation, 299
mesostructure of materials, 177
meta-mipmaps, 321–323
METs (multi-element textures), 152
microprocessor performance, 463–464
midpoint shadow maps, 278
midtones, lookup tables for, 384–385
Mie scattering, 255–259
MIMD (multiple-instruction, multiple-data)
    processors, 523, 548
mipmap-level measurements, 437–438
  derivative instructions in, 447
  emitting modified draw calls in, 443–444
  magnification in, 445
  mip-level velocity, 445–447
  multiple, 440–441
  overhead in, 444
  pixel counting in, 439–443
  results of, 441–443
  samples, 447–448
mipmap pyramids for geometry clipmaps, 28
mipmapping
  in antialiasing, 424
  in painting on meshes, 606–607
  in third-order texture filtering, 320–323
  in tile-based texture mapping, 197–198
mixed derivatives, 325
Monte Carlo methods
  for light sources, 630–632
  for soft-edged shadows, 274
Moore, Gordon, 458
Moore's Law, 458
Motion, 393
  alpha fringes in, 402–405
  bilinear filtering in, 400–405
  color transformations in, 394–395
  conditional execution in, 395
  CPU fallback in, 396–397
  debugging in, 406–407
  design for, 393–394
  high-precision storage in, 405–406
  independent neighborhood operations in, 395
  languages in, 396
  resource limits in, 397–399
  sequential neighborhood operations in, 395
  softening in, 400–402
  vertex components in, 400
motion blur removal, 434
mouse heart, MRI for, 778–779
moving instances in geometry instancing, 56
moving light in deferred shading, 149
MRI (magnetic resonance imaging), 765–766
  for human head, 779–780
  for mouse heart, 778–779
  operation of, 776–778
MRTs (multiple render targets)
  for deferred shading, 146, 153
  in hair generation, 377
multi-element textures (METs), 152
multidimensional arrays, 528–529
  1D, 529–530
  2D, 530–531

multidimensional arrays *(continued)*
3D, 531–534
higher dimensions, 534
multipass shadow algorithm, 226–228
multiple-instruction, multiple-data (MIMD) processors, 523, 548
multiple lines, compositing, 355–356
multiple parallel projection, 623–624
multiple render targets (MRTs)
for deferred shading, 146, 153
in hair generation, 377
multisample antialiasing, 481
multistreaming, 75–77
with DirectX 9.0, 78–81
implementing, 77–78
rendering in, 89
for resource management, 81–83
vertices in, 76, 83–89
multisurface pbuffers, 542

N

N^3-trees. *See* octrees
Nalu demo. *See* hair animation and rendering
NaN value, 399
nature. *See* virtual botany
Navier-Stokes equations, 717
nearest-neighbor interpolation for lookup tables, 385
nearest-neighbor sampling in texture filtering, 314–315
nearest-neighborhood filtering in Motion, 401
neighbor textures in octrees, 611
neighborhood operations in Motion, 395
nested loops for rasterization, 562
next shooters in global illumination, 643
NMR (nuclear magnetic resonance) spectroscopy, 695–696
nodes
in coherent hierarchical culling, 102
in hierarchical stop-and-wait() function, 95–96
in octrees, 597–598
noise
for alpha transparency, 14
for geometry clipmaps, 42
for grass layer, 17
Perlin. *See* Perlin noise
in shadows, 274
in sketchy rendering, 246, 248–249
nonuniformly sampled lattices, 391
nonvisible edges, extracting, 241
normal buffers in G-buffer optimization, 152–153
normal maps
for geometry clipmaps, 42, 44
in refraction simulation, 296
shading with, 120–121
normal shaders in tessellation algorithm, 118
normalization hardware, 487
normals
in ambient occlusion computations, 225, 227
in Cg vertex program, 206
nuclear magnetic resonance (NMR) spectroscopy, 695–696
null records
in filtering, 574
in sum scans, 575–576
NV_float_buffer extension, 769
NV_fragment_program2 extension, 553
NV_occlusion_query extension, 91, 93
NV_texture_rectangle extension, 769
NVPerfHUD tool, 569
NVShaderPerf tool, 569
nylon material, 184
Nyquist's rule, 423

O

object IDs in deferred shading, 146
obstacle boundaries, 757–759
occlusion, ambient. *See* ambient occlusion
occlusion culling, 93–94, 157
occlusion queries
data-dependent looping with, 554–555
in GeForce 6 Series GPU, 480, 482
hardware. *See* hardware occlusion queries
oceans. *See* water rendering
octaves in water surface models, 284
octrees, 595–596
accessing, 599–601
converting into standard 2D textures, 607–610
creating, 603–604
definition, 597
implementing, 598
indices for, 602

octrees *(continued)*
- optimizations, 602
- painting on meshes. *See* painting on meshes
- in segment buffering, 71
- storage of, 598–599
- for surface simulation, 611–612

odd-even sorts
- merge, 736–738
- transition, 734

off-chip communication in stream programming, 466
off-middle diagonals in banded sparse matrices, 707
off-screen buffers, 333, 336, 339
offsets
- for lookup tables, 387
- for multidimensional filtering, 318
- for soft-edged shadows, 278

omni lights in deferred shading, 154
omnidirectional shadow mapping, 154
one-dimensional lookup tables, 382–383
one-dimensional packing, 193
one-dimensional textures
- cubic filtering of, 318–319
- for rotationally invariant kernels, 422

one pass of sorting networks, 734
one-to-one mappings for feature vectors, 661
opacity shadow maps, 376–377, 379
OpenGL Architecture Review Board, 542
optical depth in out-scattering equation, 256
optimization, 557
- computational frequency, 561–563
  - inner-loop branching, 566
  - lookup table precomputation, 564–565
  - loop invariant precomputation, 563–564
  - in performance, 541
  - swizzle operator, 566–568
- data-parallel computing, 557–561
- for deferred shading, 147–154
- GeForce 6 Series GPU, 490–491
- in hardware occlusion queries, 105–106
- for lookup tables, 387–388
- for mipmap levels, 438
- for octrees, 602
- for Perlin noise, 415
- profiling and load balancing for, 568–570
- resource management, multistreaming for, 75–77, 81–83
  - with DirectX 9.0, 78–81
  - implementing, 77–78
  - rendering in, 89
  - vertices in, 76, 83–89
- for virtual botany, 10
- in water rendering, 288–292

options pricing, 719
- Black-Scholes model, 721–724
- lattice models in, 725–730
- option types in, 719–721

orientation for mipmap levels, 438, 447–448
oriented-disc form factor, 637
out-scattering equations, 256–257
out-slice periodic boundary faces, 756
outdoor environments
- atmospheric scattering. *See* atmospheric scattering
- nature scenes. *See* virtual botany

outflow boundaries, 756–757
outputs in Motion, 394
over-tessellation, 111–112
overdraw in deferred shading, 145
overestimated conservative rasterization, 678–680
overhead
- in hierarchical stop-and-wait function, 97
- in mipmap-level measurements, 444

overlapping objects
- in ambient occlusion computations, 227
- in collision detection, 580–581

oversampling, artifacts from, 423
overshoots in distance-mapping algorithm, 127

P

Pacific Fighters game, 283–285
packing
- in bilinear filtering, 288–290
- in data-level parallelism, 560–561
- in deferred shading, 155
- in LBM, 751–752
- in Motion, 406
- for pbuffers, 543
- texture tiles, 192–194
- for vectorization, 699–700

padding in antialiased rasterization, 336–337
painting on meshes, 602–603
- linear interpolation in, 605–606
- mipmapping in, 606–607

painting on meshes *(continued)*
  octrees in
    converting, 607–610
    creating, 603–604
  painting process, 604
  rendering, 604–607
panoramas, image, 658–660
parallax mapping
  in bump mapping, 125
  in deferred shading, 155
parallel clustering, 621
parallel computer vision processing, 664
parallel data structures
  memory models, 524–528
  multidimensional arrays, 528–534
  performance considerations, 540–543
  sparse, 535–536
    dynamic, 537–540
    static, 536–537
  streams, 521–523
  structures, 534–535
parallel hardware, 463
parallel reductions, 502, 505
parallelism
  for arithmetic intensity, 494–495
  challenges in, 468
  in computation efficiency, 462
  for Floyd-Warshall algorithm, 697
  for grids, 495–496
  importance of, 493–494
  in optimization, 557–561
  in options pricing, 728
  for stream communication, 496
parameters
  in Cg vertex program, 205–206
  constant, 505
parametric derivatives, 206
parsing command, 474
Parthenon renderer, 616–617, 631
partial derivatives, 325
partial differential equations (PDEs), 703
  Crank-Nicholson scheme for, 716–717
  sample, 714–716
  for spatial grids, 549–550
partition glass, 303–304
partitioning clipmap rings, 33–34
pass-through vertex shaders, 225
patches
  resampling, 117
  in sketchy rendering, 245
  in subdivision surfaces, 112–114
  in tessellation algorithm, 114
path tracing, 617
pbuffers (pixel buffers), 541–543
PCF (percentage-closer filtering), 269, 273, 279–280
PCI Express slots, 471–472
PDEs (partial differential equations), 703
  Crank-Nicholson scheme for, 716–717
  sample, 714–716
  for spatial grids, 549–550
pearl configuration, 368
per-hair radii, 220
per-pixel displacement mapping. *See* distance functions
per-pixel random-number generators, 395
per-vertex radii, 220
percentage-closer filtering (PCF), 269, 273, 279–280
performance
  of ambient occlusion computations, 228–230
  challenges in, 468
  computational frequency in, 541
  in deferred shading, 145, 150
  dependent texture reads in, 540–541
  of filtering, 579
  of GeForce 6 Series GPU, 489, 491
  of global illumination, 627, 645
  high-performance computing, 461–464
  of memory, 510–511
  optimization. *See* optimization
  of pbuffers, 541–543
  of Shader Model 3.0, 487–488
  of Shannon-Nyquist signal reconstruction function, 428
  of soft-edged shadows, 278–280
  of tessellation, 122
periodic boundaries, 756
Perlin noise, 409
  for alpha transparency, 14
  for grass layer, 17
  implementing, 411–415

Perlin noise *(continued)*
- optimizing, 415
- in sketchy rendering, 246, 248–249
- storage vs. computation for, 410–411

perturbations
- in refraction simulation, 296–298
- in sketchy rendering, 246–247
- in water rendering, 292–293

phase function in atmospheric scattering, 256

Phenomena renderer, 201–202
- implementing. *See* Cg vertex program
- and shaders, 202–205

Phong models
- diffuse environment maps for, 169
- in material rendering, 181–182

photon maps
- in deferred shading, 163
- in global illumination, 617–618
- as sparse structures, 537

photorealistic images, 177

piezoelectric elements, 780

ping-pong buffering, 698

pipelines
- branching in, 275, 549–550
- challenges in, 469
- in dynamic branching, 275
- flow control in, 548
- in GeForce 6 Series GPU, 476–477, 491
- in Motion, 404–405
- for rasterization, 562
- in Shader Model 3.0 programming, 487
- in stream programming, 465–467

pixel buffers (pbuffers), 541–543

pixel cells in conservative rasterization, 678–679

pixel counting in mipmap-level measurement, 439–443

pixel engines, 478

pixel pipelines, 476

pixel-processing power, 125

pixel shaders, 497
- branching in, 269, 275–276
  - implementation details, 277–280
  - performance of, 278–280
  - prediction and forecasts in, 276–277
- in dynamic irradiance environment maps, 174–175
- for geometry clipmaps, 28, 38–39, 41
- in material rendering, 183–184
- for texture access, 125

plan views for architecture, 244

plants. *See also* virtual botany
- grids for, 8–9
- strategy for, 9–10

point illumination
- light shaders for, 213
- in material rendering, 181

pointers for octrees, 598

points
- in GeForce 6 Series GPU, 475
- rendering, 518

Poisson-pressure equation, 717

pollution, atmospheric scattering from, 256

polygon edges with hemispherical projection, 641

polygons
- antialiasing, 357
- bounding, 679–686
- converting to surface elements, 223–225
- texture, 417, 438

popping in global illumination, 630–632

portal culling, 93

porting collision detection, 581–583

position encoding, 151–152

position parameters in Cg vertex program, 206

post-processing
- in deferred shading, 147
- in virtual botany, 22–24

post-projection space, 151–152

potentially visible sets (PVSs), 149

power, trends in, 460–461

power management challenges, 468–469

PPI (pulse plane-wave imaging), 780–781

precision in Shader Model 3.0 programming, 485

precomputation
- for branching, 550
- lookup tables for, 564–565
- of loop invariants, 563–564

predication for flow control, 549

prediction in soft-edged shadows, 276–277

prefiltered lines, 345
- bandlimiting signals for, 347–349
- compositing, 355–356

prefiltered lines *(continued)*
examples, 356–358
fat lines, 355
preprocessing for, 349–351
runtime, 351–355
setup for, 352–353
sharp lines, 345–348
table lookups for, 353–355
pressed wood material, 184
pricing, options, 719
Black-Scholes model, 721–724
lattice models in, 725–730
option types in, 719–721
priority queues, 104
procedural texture coordinates, 215
processors
fragment, 497–498, 770–775
for stream computation, 466–467
trends in, 458
vertex, 497–498, 526, 772–775
profiling, 568–570
program specialization feature, 530, 541
programmability, challenges in, 469
programmable parallel processors, 497–500
programmable stream processors, 467
programmable vertex engines, 478
programming models, 463
progressive refinement radiosity. *See* global illumination
projective images, 148
propagation in flow simulation, 749
protein structure prediction, 695–696
data texture indexing in, 698–699
dynamic updates in, 698
experimental results in, 701
Floyd-Warshall algorithm and distance-bound smoothing, 697
GPU implementation, 698–700
triangle approach in, 699
vectorization in, 699–700
pseudo-probability for option pricing, 725
pseudorandom values for Perlin noise, 411
PTC code for geometry clipmaps, 42
pull up in coherent hierarchical culling, 98–99
pulse plane-wave imaging (PPI), 780–781
Purcell's ray acceleration structure, 536–537
pure-z passes in multistreaming, 76
PVSs (potentially visible sets), 149

Q

quad-linear filters, 482
Quadric Error Metric algorithm, 157
quads
for computer vision, 650
in GeForce 6 Series GPU, 476
for subdivision, 111
quadtrees, 598
in adaptive subdivision, 644
in global illumination, 644–645
quality
in deferred shading, 154–158
in G-buffer optimization, 153
in water rendering, 288–292
quasi-Monte Carlo sampling, 630–632
quasi-optimal antialiasing, 424–427
queries, hardware occlusion. *See* hardware occlusion queries
queues
in mipmap-level measurements, 444
priority, 104
Quicksort algorithm, 734, 736

R

radial distortion in computer vision, 651–653
radio frequency (RF) in MRI, 777
radiosity in global illumination, 617, 635–637
adaptive subdivision in, 643–645
form factor computation in, 641–643
foundations of, 636–638
GPU implementation, 638–639
next shooters in, 643
progressive refinement in, 637–638
quadtree subdivision in, 644–645
visibility using hemispherical projection, 639–641
random-number generators, 395
random sparse matrices, 708
random sparse matrix-vector product, 711–712
randomly placed samples for soft-edged shadows, 274
randomness, noise for, 409
RAPID collision-detection scheme, 579
rasterization, 616–617
antialiased. *See* antialiased rasterization

rasterization *(continued)*
computation efficiency in, 462
conservative. *See* conservative rasterization
CPU-GPU analogies, 501
in GeForce 6 Series GPU, 481
in global illumination. *See* global illumination
in occlusion culling, 93
pipelines for, 562
rasterization blocks, 476
rasterizers
for FFT, 772–775
in parallel processors, 498
Rational Quantify tool, 569
ray casts
as multiple parallel projection, 623–624
in virtual botany, 21
ray tracing
in final gathering, 618–620
in view-dependent displacement mapping, 126
Rayleigh scattering, 255–259
reaction-diffusion model, 505–508
read-only vertex buffers, 30–31
readback in fast algorithms, 512–513
reads, dependent texture, 540–541
real spherical harmonic transforms, 170
real-time atmospheric scattering, 258–260
real-time computation of dynamic irradiance environment maps. *See* dynamic irradiance environment maps
real-time global illumination, 163–164
real-time optimization, 10
real-time rendering, 182–184
real-time volumetric shadows, 375–378
receivers in ambient occlusion computations, 225–226
reconstruction
in deferred filtering, 671–673
derivative, 324–327
in global illumination, 639
image, 427–430
rectangle textures, 529
recursive cubic convolution, 315–320
reductions
CPU-GPU analogies, 502–503
in stream programming, 464
reflectance models for hair, 369–375
reflection maps, 303
reflections
in deferred shading, 148
in indirect lighting, 231
reflective surfaces and Phenomena, 203
refraction simulation, 295–296
glass simulation, 303–305
masks in, 297–300
techniques, 296–297
water simulation, 300–302
region-of-interest decoding, 42
regions
for geometry clipmaps, 28–29
in sketchy rendering, 246
registers
in GeForce 6 Series GPU, 491
in Shader Model 3.0 programming, 483
swizzle operator for, 566
VPOS, 173
relief mapping, 126
render contexts
in geometry instancing, 50–51
for pbuffers, 542
render phase in geometry instancing, 53
render targets in Shader Model 3.0 programming, 484–485
render-to-texture feature, 499
CPU-GPU analogies, 501
for deferred filtering, 671
in dynamic irradiance environment maps, 175
in water rendering, 291
render-to-vertex arrays, 225
render-to-vertex-buffer operation, 525
rendering
blueprints. *See* blueprints
geometry clipmaps, 32–39, 43
in global illumination. *See* global illumination
hair, 220
for materials, 181–184
in multistreaming, 89
painting on meshes, 604–607
points, 518
sketchy. *See* sketchy rendering
terrain. *See* geometry clipmaps
in tile-based texture mapping, 198
water. *See* water rendering
repeated edges, 248
resampling
images, 422–423

resampling *(continued)*
  patches, 117
residual energy values in progressive refinement, 637–638
residual textures in global illumination, 638
residuals for geometry clipmaps, 42
resolution
  in global illumination, 645
  for mipmap levels, 438
  in octrees, 597
  for painting on meshes, 603
resource limits in Motion, 397–399
resource management, multistreaming for, 75–77, 81–83
  with DirectX 9.0, 78–81
  implementing, 77–78
  rendering in, 89
  vertices in, 76, 83–89
result parameters with Phenomena, 204
RF (radio frequency) in MRI, 777
RGB calibration data in mipmap-level measurements, 444
RGBA texels, LBM, 751
rings, clipmap, 33–34
riskless rate of return of options, 720
robustness in conservative rasterization, 686–687
root shaders, 204
roots of octrees, 597
rotationally invariant kernels, 1D textures for, 422
roughened profiles, 247–248
round-robin queues, 444

S

sampling
  artifacts from, 346, 423
  centroids, 485
  in distance-mapping algorithm, 128
  in G-buffer optimization, 152–153
  in global illumination, 627
  height maps, 286–287
  for light sources, 630–632
  for PCF of shadow maps, 279–280
  for soft-edged shadows, 274–278
  in three-dimensional lookup tables, 384
  in water rendering, 286–287
saturation
  in HSV color space, 656
  lookup tables for, 384–385
scalability in deferred shading, 154–155
scalable vertex processing, 475
scale functions in atmospheric scattering, 261
scaling
  distances in, 125
  for lookup tables, 387
scans, sum, 574–576
scatter and scattering, 515
  address sorting in, 516–518
  atmospheric. *See* atmospheric scattering
  gather operations from, 515–516
  in Motion, 394
  rendering points in, 518
  through search/gather, 575–577
  in sparse data structures, 537, 539
  in stream communication, 496
scattering coefficients, FFT for, 781
Scholes, Myron, 721
scissor tests, 479–480
screen-door effect, 13–15
screen resolution for mipmap levels, 438
screen space stencil masking costs, 155
searches
  for displaced values, 577–578
  in relief mapping, 126
  vs. sorts, 579
second pass in global illumination, 629
sector-portal structures, 157
segment buffering, 69
  improving, 72
  methods of, 70–72
  problem space in, 69–70
  solution, 70
segments in hair generation, 368
semidiagonals in conservative rasterization, 679
separable filters, 338–339
sequences of fragment programs, 651–655
sequential Floyd-Warshall algorithm, 697
sequential neighborhood operations, 395
shade maps in sketchy rendering, 245–246
Shader Model 3.0 programming, 483
  performance of, 487–488
  pixel shaders in, 484–486
  vertex shaders in, 483–484

shader pipelines
  in GeForce 6 Series GPU, 491
  in Shader Model 3.0 programming, 487
shaders
  in GeForce 6 Series GPU, 477
  for lookup tables, 386–388
  and Phenomena, 202–205
  pixel. *See* pixel shaders
  vertex. *See* vertex shaders
shading
  deferred. *See* deferred shading
  for geometry clipmaps, 38–39
  hair, 369–378
  with normal mapping, 120–121
shading maps, 179–181
shadow acne, 278
shadow buffers, 482–483
shadow casting in Cg vertex program, 211
shadow maps
  in Cg vertex program, 220
  in deferred shading, 149–150
  in hair generation, 376
  light shaders for, 213
  for opacity, 376–377, 379
  for soft-edged shadows, 276
  variables for, 212
  z-buffers for, 270
shadow volumes, 270
shadows
  ambient occlusion for, 225–230
  lookup tables for, 384–385
  optimizing, 153–154
  soft-edged. *See* soft-edged shadows
  in virtual botany, 20–22
shallow-water wave equation, 714
Shannon-Nyquist signal reconstruction function, 427–430, 432
shared buffers, 474
sharp lines, 345–348
shock filtering, 430–432
shower-door effect, 249–251
shrub layers, 18–20
sigma values in hair generation, 377
signs in floating-point representation, 513–514
silhouettes for prefiltered lines, 357
SIMD (single-instruction, multiple-data) operation
  branching in, 548
  for computer vision, 649
  for fragment processors, 523
  in GeForce 6 Series GPU, 476
sinc filters, 351, 427
single float representation in linear equations, 704
single-instruction, multiple-data (SIMD) operation
  branching in, 548
  for computer vision, 649
  for fragment processors, 523
  in GeForce 6 Series GPU, 476
single shadow maps, soft-edged shadows with, 271–280
size
  of geometry clipmaps, 31–32
  trends in, 458
sketchy drawings, 235–236
  depth sprite rendering in, 237–238
  edge enhancement in, 236–237
  intermediate rendering results for, 236
sketchy rendering, 244
  depth adjustments in, 247
  edges and color patches in, 245, 248
  shower-door effect in, 249–251
  uncertainty in, 245–249
  variations of, 247–248
skinning in geometry instancing, 56
sky boxes in deferred shading, 150
sky dome blooming, 22–23
smoothness
  in filter degrees, 315
  noise for, 409
sniper-scope lenses, refraction for, 295
soft-edged shadows, 269
  blurring hard-edged shadows for, 271–274
  branching in, 275–276
  current techniques, 270–271
  efficiency in, 274–277
  implementation details, 277–280
  performance of, 278–280
  prediction and forecasts in, 276–277
soft shadows in virtual botany, 21
softening in Motion, 400–402
solid geometry in Marschner reflectance models, 375
sorting
  addresses, 516–518
  algorithms for, 733–734

- sorting *(continued)*
  - bitonic merge sorts, 742
    - algorithm for, 739
    - grouping keys in, 740–741
    - implementing, 743–744
  - fast sorting, 735
  - odd-even merge sorts, 736–738
  - resources for, 738–742
  - vs. searches, 579
  - simple approach to, 734–735
- sparse data structures, 535–536
  - dynamic, 537–540
  - static, 536–537
- sparse matrices
  - banded, 706–707, 710–711
  - random, 708
- sparse matrix-vector products, 710–711
- spatial image position in three-dimensional lookup tables, 384
- spatial organization in segment buffering, 71
- spatial scales in water surface models, 285
- spatial to frequency domains, 172–173
- specialization in computation efficiency, 462
- specular color, 80
- specular irradiance environment maps, 169
- specular reflections, 148
- speed, trends in, 458
- SpeedTreeCAD program, 18
- SpeedTreeRT program, 13–14, 18
- sphere tracing
  - in displacement mapping, 126
  - in distance-mapping algorithm, 128–129
- spheres in hair generation, 367–368
- spherical harmonic convolution, 170–172
- spotlight illumination, 213
- spotlight shaders, 211
- stages of sorting algorithms, 736
- stained glass, 303–304
- *S.T.A.L.K.E.R.*, deferred shading in. *See* deferred shading
- stalls
  - in coherent hierarchical culling, 103
  - in hierarchical stop-and-wait function, 96
- standard rasterization, 677–678
- static batching, 54–56, 66
- static branch resolution, 549–550, 566
- static sparse structures, 536–537
- statistics in Motion, 394
- steady-state responses for filters, 419
- stencil buffers, 480
- stencil masks, 150, 155
- stencil routing, 537
- stencil shadow volumes, 270
- stereographic projection, 639–641
- stochastic differential equations, 721
- stochastic sampling, 335
- stone surface rendering, 123–124
- storage
  - in computation efficiency, 462
  - formats for, 488–489
  - in G-buffer optimization, 153
  - in Motion, 405–406
  - for noise, 410–411
  - of octrees, 598–599
- strata for soft-edged shadows, 274
- stratified sampling, 274
- stream communication, 496
- stream computation, 464
  - processors for, 466–467
  - programming model for, 464–466
- stream reduction operations, 573
  - collision detection, 579–583
  - filtering
    - through compaction, 574–579
    - for subdivision surfaces, 583–587
- streaming architectures, 457
- streams, 521–523
  - for batching, 63
  - CPU-GPU analogies, 500
  - in flow simulation, 749
  - in GPGPU framework, 504
  - in LBM, 752–753
  - multistreaming. *See* multistreaming
  - types of, 525–526
- strike price of options, 720
- structures, 534–535
  - sparse, 535–536
    - dynamic, 537–540
    - static, 536–537
  - streams of, 534–535
- subdivision curve interpolants, 40
- subdivision
  - adaptive, 112
    - in global illumination, 643–645
    - surfaces in, 584
    - texture quadtrees in, 644

subdivision *(continued)*
surfaces, 109–110
Catmull-Clark, 110–111
definitions for, 110
displacement mapping for, 119–121
filtering for, 583–587
patching, 112–114
and tessellation, 111–119
subpixels
filtering, 419–420
Gaussian, 420–421
substreaming, 566
substructuring, 645
suede material, 184
sum scans, 574–576
summation operations in computer vision, 655–658
sun
in deferred shading, 149–150
in dynamic irradiance environment maps, 175
for grass layer, 15–16
supersampling rates, 331
superscalar instructions, 568
surface-light interaction in deferred shading, 157–158
surface-painting applications, 596
surface-scattering equations, 257–258
surfaces
in derivative reconstruction, 326
distance maps of, 128
in dynamic ambient occlusion, 223–225
pbuffer, 542
simulation, 611–612
subdivision. *See* subdivision
in water rendering, 284–285
sweaters, 184–186
swizzle operator, 559, 566–568
synthesis for geometry clipmaps, 29, 40
system architecture, 471–473
fixed-function features in, 481–483
graphics operations, 473–477
non-graphics operations, 478–480
system memory, 474

T

T-junctions in watertight tessellation, 119
T streams in multistreaming, 76–78, 84
table lookup. *See* lookup tables (LUTs)
target switches in deferred shading, 155
task parallelism
in computation efficiency, 462
in stream programming, 467
Taylor expansion, 714
technology trends, 457, 468
Temple of Ramses II, 244
temporal coherence, 97–98
tensor-product version for curve interpolants, 40
terrains
alpha tests in, 440
geometry clipmaps for. *See* geometry clipmaps
in mipmap-level measurements, 447–448
tessellation
adaptive, 109
algorithm for, 114–119
in deferred shading, 157
in hair generation, 364–365
performance of, 122
subdivision for, 111–113
watertight, 118–119
texel pipelines, 476
texture-aliasing artifacts, 423
texture caches, 475
texture contexts, 50–51
texture indirections
in Motion, 398
in octrees, 598
texture load/filter engines, 478
texture-lookup shaders, 215
texture lookups for filters, 418
texture maps
lookup tables for, 564–565
in multistreaming, 76
tile-based. *See* tile-based texture mapping
texture polygons
for filters, 417
for mipmap levels, 438
texture quadtrees, 644
texture shaders
in Cg vertex program, 215–216
and Phenomena, 202
texture tiles, 189–191
constructing, 191–192
mapping, 195
packing, 192–194
texture units, 476, 478, 490, 498

- textures
  - bindings of, 662
  - coordinates for
    - in Cg vertex program, 206
    - CPU-GPU analogies, 501–502
    - for filters, 418–419
    - in Motion, 400
    - in multistreaming, 80
    - precomputation of, 563
    - in protein structure prediction, 698–699
    - in recursive cubic convolution, 317
    - in sketchy rendering, 250
    - swizzle operator for, 567
  - CPU-GPU analogies, 500
  - filtering, 192
    - adaptive, 434
    - in loading, 490
    - for mipmap levels, 438
    - third-order. *See* third-order texture filtering
    - types of, 314–315
  - in GeForce 6 Series GPU, 474, 482
  - for geometry clipmaps, 44
  - lookup, 317, 327–328
  - memory footprints for, 322
  - in multistreaming, 86
  - pixel shaders for, 125
  - in Shader Model 3.0 programming, 484
  - streams for, 526
  - for water rendering. *See* water rendering
  - wrapping, 42
- third-order texture filtering, 313–315
  - derivative reconstruction in, 324–327
  - mipmapping in, 320–323
  - recursive cubic convolution in, 315–320
- three-dimensional arrays, 531–534
- three-dimensional lookup tables, 383–385
- three-dimensional surfaces, 538
- tile-based texture mapping, 189–190
  - mipmap issues in, 197–198
  - process, 195–196
  - tile construction in, 191–192
  - tile packing in, 192–194
- tiles
  - in antialiased rasterization, 331–333, 339
  - in sparse data structures, 538–539
- time scales in water surface models, 285
- time warping, 433–434
- tone mapping, 162
- toroidal access for geometry clipmaps, 39–40
- tracking hands in computer vision, 656–658
- transistors
  - challenges in, 468
  - in computation efficiency, 462
  - trends in, 458
- transition regions for geometry clipmaps, 28–29
- translation, address, 530–534
- translucent cylinders, 371
- translucent surfaces, 148
- transparency
  - in deferred shading, 162
  - simulating, 13–15
- transparent materials, 221
- trapezoid rule, 254
- traversing
  - nodes in coherent hierarchical culling, 102, 104–105
  - triangle lists, 537
- tree layers, 18–20
- tree lookup for octrees, 599–601
- trees
  - in hierarchical stop-and-wait function, 95
  - modeling. *See* virtual botany
  - in segment buffering, 71
- trends
  - compute vs. communicate, 459–460
  - consequences of, 458
  - core technology, 458
  - GPU functionality subsumed by CPUs, 470
  - latency vs. bandwidth, 459
  - in power, 460–461
  - power management, 468–469
  - programmability and functionality, 468–469
- triangles
  - computation efficiency in, 462
  - in conservative rasterization, 683–684
  - in GeForce 6 Series GPU, 475
  - for hair rendering, 220
  - overlapping, 580–581
  - in protein structure prediction, 699
  - in subdivisions, 585–586
- tricubic filters, 317, 324
- trigonometric functions, 484
- trilinear filters
  - in GeForce 6 Series GPU, 482
  - in water rendering, 288–290
- trilinear interpolation, 385–386

trilinear reconstruction filters, 324
trilinear texture lookups, 317
two-dimensional arrays, 530–531
two-dimensional isosurfaces, 538
two-dimensional packing, 193
two-dimensional reflection maps, 303
two-pass methods in gathering, 617–621
two-pass separable filters, 338–339

U

ultrasonic imaging, 780–783
ultrasonic scanning, 765
uncertainty in sketchy rendering, 245–249
under-tessellation, 111
underestimated conservative rasterization, 678–680, 686
underlying assets of options, 720
undersampling, artifacts from, 423
uniform arguments, computational frequency of, 541
uniform tessellation, 113
unrolled cube-map faces in deferred shading, 154
update phase in geometry instancing, 53
update rate for geometry clipmaps, 43
updates
  for geometry clipmaps, 39–43
  in protein structure prediction, 698
upsampling for geometry clipmaps, 40–41

V

valences for vertices, 110
value functions, 504
value in HSV color space, 656
variation in grass layer, 15
VBO (vertex buffer object) routines, 343
vector-vector sums, 709
vectors
  for computer vision, 661–663
  in linear equations, 704–705, 709–710
  in Marschner reflectance models, 374
  for Perlin noise, 411
  in protein structure prediction, 699–700
Verlet integration, 366
vertex buffer object (VBO) routines, 343
vertex buffers
  binding to streams, 81
  for geometry clipmaps, 30–31, 33–34
  for geometry instancing, 58, 62
  for grass layer, 12–13
  in multistreaming, 85–88
vertex constants instancing, 57–61
vertex fetch units, 474
vertex processors
  characteristics of, 497–498
  for FFT, 772–775
  for fragment streams, 526
  in GPUs, 125, 475
vertex programs
  Cg. *See* Cg vertex program
  in GeForce 6 Series GPU, 475
  in stream programming model, 467
vertex shaders, 497
  in atmospheric scattering, 262–264
  for distance functions, 130
  in GeForce 6 Series GPU, 478
  for geometry clipmaps, 28, 36–38
  for geometry instancing, 60, 64
  for grass layer, 16
  in material rendering, 183–184
  in Shader Model 3.0 programming, 483–484
  in water rendering, 287
vertex shading units, 474–475
vertex streams, 525
  computational frequency of, 541
  in GeForce 6 Series GPU, 481
vertex texture fetch (VTF) feature, 498
vertex textures
  for geometry clipmaps, 44
  in Shader Model 3.0 programming, 484
  for water rendering. *See* water rendering
vertices
  in conservative rasterization, 682–683
  in CPU-GPU analogies, 502
  for feature vectors, 662
  in GeForce 6 Series GPU, 474
  in geometry instancing, 57–61
  in geometry packets, 49
  in Motion, 400
  in multistreaming, 76, 80, 83–89
  for subdivision, 111
  for surfaces, 110
  in tessellation algorithm, 118
  in voxelization, 755–756
VideoOrbits algorithm, 658–660

view-dependent displacement mapping, 126
view frustum culling, 35
view space in G-buffer optimization, 151
viewer motion with geometry clipmaps, 39
virtual botany, 7
  grass layer, 11–17
  ground cover for, 17–18
  planting grid for, 8–9
  planting strategy in, 9–10
  post-processing in, 22–24
  real-time optimization for, 10
  scene management in, 7–8
  shadows in, 20–22
  tree and shrub layers, 18–20
virtual positions
  in deferred shading, 155–156
  in G-buffer optimization, 152
virtual shadow depth cube textures, 154
visibility
  in coherent hierarchical culling, 98, 100
  in deferred shading, 157
  in hardware occlusion query optimizations, 105–106
  hemispherical projection for, 639–641
  ray casts for, 21
visibility pass for global illumination, 639
visibility term for radiosity form factor, 636
visible edges, extracting, 241
visible points data, 628
vision, computer. *See* computer vision
visualization in flow simulation, 759–760
volume rendering, gradients in, 324
volume shaders
  in Cg vertex program, 217–218
  and Phenomena, 202–203
volume texture indexing, 157
volumetric fog, 146
volumetric shadows, 375–378
voxelization of boundaries, 748, 754–756
VPOS input register, 173
VTF (vertex texture fetch) feature, 498

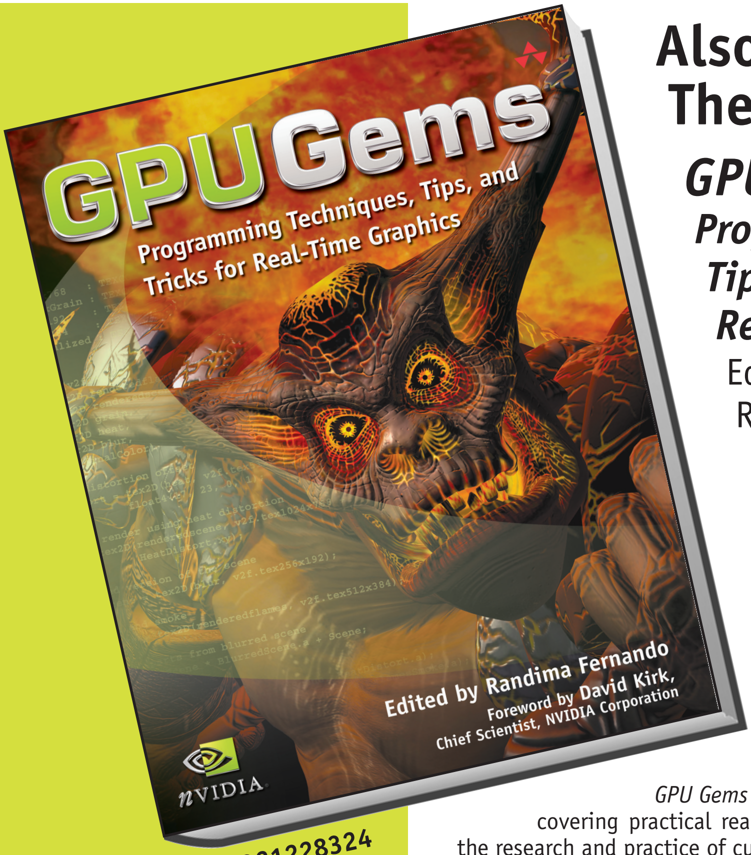
W

Wang tiles, 191, 193
water rendering, 283
  analytical deformation models in, 292–293
  back side of waves in, 291–292
  branching in, 290
  dynamic displacement mapping in, 293
  foam generation in, 293
  height map sampling in, 284–287
  implementation details, 285–286
  local perturbations in, 292–293
  models for, 283–284
  quality improvements in, 288–292
  refraction for. *See* refraction simulation
  render-to-texture in, 291
  surface models in, 284–285
watertight tessellation, 118–119
wavelengths in atmospheric scattering, 257
wax material, 184
weight functions, 318
weight maps, 369
weights for recursive cubic convolution, 316–318
`WGL_ARB_pbuffer` extension, 769
`WGL_ARB_render_texture` extension, 769
wind for grass layer, 17
window space in conservative rasterization, 681
wool material, 184–186
wrapped hemispherical sources, 146
write mask operators, 567

X · Y · Z

X-PLOR package, 695
x scissoring, 479
y scissoring, 479
z-buffers
  in GeForce 6 Series GPU, 482
  for shadow maps, 270
z-compare operations, 477
z-cull units, 479–480
z-culls, in GeForce 6 Series GPU, 479–480, 482, 490
z-only rendering pass, 105
z-scale values in flatness tests, 115
z-values in sketchy rendering, 247
zero, division by, 399
zero-alpha pixels, 404
zeroth-order linear filtering, 314–315

Pharr_bbad.qxd 1/26/05 1:50 PM Page 1


GPU Gems
Programming Techniques, Tips, and Tricks for Real-Time Graphics
Edited by Randima Fernando
Foreword by David Kirk, Chief Scientist, NVIDIA Corporation
NVIDIA

0321228324

Also Available— The First Volume

GPU Gems: Programming Techniques, Tips, and Tricks for Real-Time Graphics

Edited by Randima Fernando

Winner of the 2004 Front Line Award for Best Book from *Game Developer Magazine*

GPU Gems is a compilation of contributed chapters covering practical real-time graphics techniques arising from the research and practice of cutting-edge developers. It focuses on the programmable graphics pipeline available in today's GPUs and highlights quick and dirty tricks used by leading developers, as well as fundamental, performance-conscious techniques for creating advanced visual effects. *GPU Gems* and *GPU Gems 2* together provide a comprehensive guide to taking advantage of the power of the new generation of GPUs.

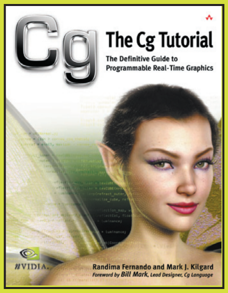
Major topics covered include:

- Natural Effects
- Lighting and Shadows
- Materials
- Image Processing
- Performance and Practicalities
- Beyond Triangles

Contributors are from the following corporations and universities:

Alias Systems
Brown University
Croteam
Cyan Worlds
Hochschule Bremen
Industrial Light & Magic
iXBT.com
Monolith Productions
New York University
NVIDIA
Paralelo Computação
Piranha Bytes
Pixar Animation Studios
Siemens Medical Solutions
Softimage Co.
Softlab-NSK
Sony Pictures Imageworks
Stanford University
UC Davis
UNC-Chapel Hill
Universitat Pompeu Fabra
University of Utah
University of Waterloo

Also of interest


Cg
The Cg Tutorial
The Definitive Guide to Programmable Real-Time Graphics
NVIDIA
Randima Fernando and Mark J. Kilgard
Foreword by Bill Mark, Lead Designer, Cg Language

The Cg Tutorial
by Randima Fernando and
Mark J. Kilgard
0321194969

NVIDIA

Addison Wesley

For additional information: http://www.awprofessional.com • http://developer.nvidia.com

Wouldn't it be great if the world's leading technical publishers joined forces to deliver their best tech books in a common digital reference platform?

They have. Introducing **InformIT Online Books** **powered by Safari.**

- **Specific answers to specific questions.**
InformIT Online Books' powerful search engine gives you relevance-ranked results in a matter of seconds.

- **Immediate results.**
With InformIT Online Books, you can select the book you want and view the chapter or section you need immediately.

- **Cut, paste and annotate.**
Paste code to save time and eliminate typographical errors. Make notes on the material you find useful and choose whether or not to share them with your work group.

- **Customized for your enterprise.**
Customize a library for you, your department or your entire organization. You only pay for what you need.

Get your first 14 days FREE!
For a limited time, InformIT Online Books is offering its members a 10 book subscription risk-free for 14 days. Visit **http://www.informit.com/onlinebooks** for details.

POWERED BY Safari® TECH BOOKS ONLINE®

informIT Online Books

informit.com/onlinebooks

informIT

www.informit.com

YOUR GUIDE TO IT REFERENCE

Articles

Keep your edge with thousands of free articles, in-depth features, interviews, and IT reference recommendations – all written by experts you know and trust.

Online Books

Answers in an instant from **InformIT Online Book's** 600+ fully searchable on line books. For a limited time, you can get your first 14 days **free**.

POWERED BY Safari® TECH BOOKS ONLINE®

Catalog

Review online sample chapters, author biographies and customer rankings and choose exactly the right book from a selection of over 5,000 titles.

Register
Your Book

at www.awprofessional.com/register

You may be eligible to receive:

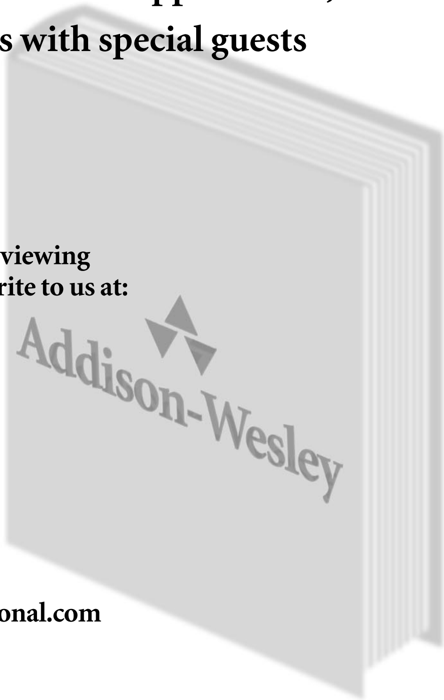
- **Advance notice of forthcoming editions of the book**
- **Related book recommendations**
- **Chapter excerpts and supplements of forthcoming titles**
- **Information about special contests and promotions throughout the year**
- **Notices and reminders about author appearances, tradeshows, and online chats with special guests**

Contact us

If you are interested in writing a book or reviewing manuscripts prior to publication, please write to us at:

Editorial Department
Addison-Wesley Professional
75 Arlington Street, Suite 300
Boston, MA 02116 USA
Email: AWPro@aw.com

Visit us on the Web: http://www.awprofessional.com



Addison-Wesley Warranty on the CD-ROM

Addison-Wesley warrants the enclosed disc to be free of defects in materials and faulty workmanship under normal use for a period of ninety days after purchase. If a defect is discovered in the disc during this warranty period, a replacement disc can be obtained at no charge by sending the defective disc, postage prepaid, with proof of purchase to:

Disc Exchange
Addison-Wesley Professional
Pearson Technology Group
75 Arlington Street, Suite 300
Boston, MA 02116
Email: AWPro@aw.com

Addison-Wesley makes no warranty or representation, either expressed or implied, with respect to this software, its quality, performance, merchantability, or fitness for a particular purpose. In no event will Addison-Wesley, its distributors, or dealers be liable for direct, indirect, special, incidental, or consequential damages arising out of the use or inability to use the software. The exclusion of implied warranties is not permitted in some states. Therefore, the above exclusion may not apply to you. This warranty provides you with specific legal rights. There may be other rights that you may have that vary from state to state. The contents of this CD-ROM are intended for personal use only, unless otherwise addressed under a separate license.

NVIDIA Statement on the Software

The source code provided is freely distributable, so long as the NVIDIA header remains unaltered and user modifications are detailed. NVIDIA makes no warranty or representation that the techniques described herein are free from any Intellectual Property claims. The reader assumes all risk of any such claims based on his or her use of these techniques.

NO WARRANTY
THE SOFTWARE AND ANY OTHER MATERIALS PROVIDED BY NVIDIA ON THE ENCLOSED CD-ROM ARE PROVIDED “AS IS.” NVIDIA DISCLAIMS ALL WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.

LIMITATION OF LIABILITY
NVIDIA SHALL NOT BE LIABLE TO ANY USER, DEVELOPER, DEVELOPER’S CUSTOMERS, OR ANY OTHER PERSON OR ENTITY CLAIMING THROUGH OR UNDER DEVELOPER FOR ANY LOSS OF PROFITS, INCOME, SAVINGS, OR ANY OTHER CONSEQUENTIAL, INCIDENTAL, SPECIAL, PUNITIVE, DIRECT OR INDIRECT DAMAGES (WHETHER IN AN ACTION IN CONTRACT, TORT OR BASED ON A WARRANTY), EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THESE LIMITATIONS SHALL APPLY NOTWITHSTANDING ANY FAILURE OF THE ESSENTIAL PURPOSE OF ANY LIMITED REMEDY. IN NO EVENT SHALL NVIDIA’S AGGREGATE LIABILITY TO DEVELOPER OR ANY OTHER PERSON OR ENTITY CLAIMING THROUGH OR UNDER DEVELOPER EXCEED THE AMOUNT OF MONEY ACTUALLY PAID BY DEVELOPER TO NVIDIA FOR THE SOFTWARE OR ANY OTHER MATERIALS.

More information and updates are available at:
http://developer.nvidia.com
http://www.awprofessional.com