Index

A, B
Accelerometer, 26, 64–67
afconvert, 91–92
afinfo, 90–91
afplay, 92
Apple Developer, 7–8
AR applications
  ARController
    ARController.m, 229–230
    initWithViewController method, 230
    location sharing, 234, 235
  RootViewController, 231
    updated AppDelegate.h, 232
    updated AppDelegate.m, 233
    updated ARController.h, 229
    updated RootViewController.h, 231
    updated RootViewController.m, 231–232
    credits, 228
  Facebook application (see Facebook application)
  Facebook iOS SDK, 229
  frameworks, 228
sensor updates
  Degrees to Radians, 235–236
  deviceOrientationDidChange method, 236–237
  Heading and Location properties, 237
  updated ARController.h, 236
  social context (see Facebook context)
  storing coordinates
    addCoordinate method, 245
angleFromCoordinate method, 241
ARAnnotation.h, 243
ARAnnotation.m, 244
ARController.h, 241
ARCoordinate property, 241
ARCoordinate.h, 238, 244
ARCoordinate.m, 238–239
ARGeoCoordinate.h, 239
ARGeoCoordinate.m, 239–240
calibrateUsingOrigin method, 241
deltaAzimuthForCoordinate method, 245–246
handling location and heading, 248
Imports and Private method, 241
initWithCoordinate method, 241
initWithCoordinateAndOrigin method, 241
isNorthForCoordinate method, 246
location and heading methods, 242
new private methods, 245
pointForCoordinate method, 246–247
setting initial values, 248
updated ARController.h, 242
updated ARController.m, 243
updated initWithViewController method, 241–242
updated updateLocations method, 247
viewportContainsCoordinate method, 245
ARKit toolkit, 181, 208
ASI-HTTP-Request Library, 300, 314–315
Audio File Convert, 91
Augmented reality (AR)
  applications (see AR applications)
  ARKit, 181, 208
  2D game building framework (see Cocos2D framework, iPhone)
  facial recognition (see Facial recognition techniques, mobile iOS environment)
  gaming and location-based AR, 4 markers
    correct orientation, 211
    EAGLView, 213
    high-contrast, unique image, 211
  String SDK, 213
  ViewController, 219
Powered by String SDK
  asynchronous callback method, 186
  Demo version, 182
  frame buffer, 185–186
  getCurrentVideoBuffer method, 185
  handleSnapshot method, 186
  initialization code, 184
  marker-based application, 183
  markers, 184
  OpenGL AR application,
    implementation steps, 184–185
    render method, 185
    resume method, 186
  screen buffer delegate, 185
  shaders and lighting effects, 186–187
Unity 3D integration, 186
QCAR demo
  creating new project, 190
  creating new trackable link, 190–192
  creating Xcode project, 192–194
  downloading trackable link, 192
  EAGLView (see EAGLView)
  redirecting UIView (see UIView)
Qualcomm, 181
Qualcomm SDK
  Dominoes application, 189
  Dominoes demo, 189
  downloading QCAR SDK, 187–188
  installation directory, 188–189
  mixed-reality space, virtual button, 189–190
  QCAR SDK, 188–189
String SDK, 181
  toolkits, 181–209
AVAudioPlayer Class, 93–94
AVFoundation Framework, 24, 194
Azimuth
  calculation, 227
  formula, 228
  orientation changes, 228
Camera capture
  front-facing camera, 101–102
  isSourceTypeAvailable method, 101–102
photo capture
  e-mailing image, 111–114
  saving images in different formats, 110–111
  using camera, 109–110
  using storyboards, 103–108
  UIImagePickerController class, 101
  UIImagePickerControllerCameraDeviceFront parameter, 102
  UIImagePickerControllerSourceType Camera, 102
CCLayer, Cocos2D, 131
Chipmunk
  cpDampedSprint joint, 156
  createBoxAtLocation method, 154
  createGround method, 153–154
  createSpace method, 153
  draw (override) method, 154
  init and update method, 155
  joints, 155
  mouse joint, 155
  scheduleUpdates method, 156
Index

Spring joint
  CCSpriteFrameCache, 158
  cpDampedSpring constraint, 158
  CPSprite menuPumpkin1, 158
  updated init method, 158
  updated update method, 159
touch events, 156–157
updated MenuLayer.h, 157
Cocos2D framework, iPhone, 299
adding camera view, 128–129
adding effects
  sound effects, 135
  touch effects, 131, 133
  visual effects, 133
adding HUD layer
  HelloWorldLayer.h, 137
  HUDLayer.h, 136
  HUDLayer.mm, init method, 136
  incrementCounter method, 136, 138
  Replace init method, 137
  Scene method, 137
adjusting default view
  changing pixelFormat, 127
  EAGLView, 127
  main and sub views, 127
  overlay view, 127
  UIView, 126
CCLayer, 131
director, 130–131
EAGLContext object, 126
installation
  creating project, 125–126
  project templates for Xcode, 124–125
OpenGL rendering, 126
scaling camera view, 129–130
scenes, 130
Cocos2D game
  applicationDidFinishLaunching method, 144
artwork
  CPSprite.h, 150
  CPSprite.m, 151–152
  particleTexture.png, 149
  PumpkinExplosion.plist, 149
  Pumpkins.plist, 149
  Pumpkins.png, 149
  pumpkin sprite sheet, 150
camera support, 163–170
camera view
  AppDelegate.h, 145
  AppDelegate.m, 145–146
  CoreImage, 145
  CoreMedia, 145
  CoreVideo, 145
  frontFacingCameraIfAvailable, 146
CCNode template, 147
Chipmunk (see Chipmunk)
CIDetector class, 141
CIDetector property, 171
cocos2d_chipmunk template, 142–143
Endlayer.h file, 177
EndLayer.m file, 177–178
facialRecognitionRequest method, 169
game menu,
  applicationDidFinishLaunching, 148
GitHub repository, 143
Grossini multiplication, 144
Hello World project, 144
helper code directory
  cpMouse.c, 152
  cpMouse.h, 152
  drawSpace.c, 152
  drawSpace.h, 152
Image-Handling methods, 165–168
import statements, 148
MenuLayer.h, 147, 153
MenuLayer.m, 147
Menu option, 160–163
newGameTapped method, 162
OpenGL ES view, pixelFormat, 164
project creation, 142–144
pumpkins
  facialRecognitionRequest method, 173–174
  init method, 172–173
  private variables, 172
Touch methods, 175
static instance method, 169
updated ActionLayer.h, 160
Updated ActionLayer.m, 161
updated AppDelegate.h, 163–164
updated facialRecognitionRequest method, 171
updated init method, 168–169
updated Private Method declaration, 168
CoreMedia.framework, 194
CoreVideo.framework, 194

D
2D gaming applications. See Cocos2D framework, iPhone; Cocos2D game
didUpdateHeading Delegate Method, 80
didUpdateHeading method, 83
Digital compass, 21–23
Director, Cocos2D, 130–131
3D lamp, 186

E
EAGLView
configuring video as background image, 204
createFramebuffer method, 197–198, 216–217
deleteFramebuffer method, 198, 217
external methods for ViewController, 199–200
Handle Status Updates, 200–201
import statements, 194, 214
init methods, 202
initializing OpenGL rendering and render frames, 205–206
initWithCoder, 197
initWithCoder method, 215
interface, 195
layerClass method, 215
layerClass static method, 196
loading marker/trackable, 201–202
loading test shaders, 204–205
new import statements, 196
new method declarations, 195
presentFramebuffer and layoutSubviews methods, 199
private methods, 196, 214
QuartzCore Library, 214
setContext method, 216
setFramebuffer method, 199, 218
starting and stopping camera view, 203
status enumeration, 195
synthesize the properties, 215
updated @interface, 214
Effects, Cocos2D
sound effects, 135
touch effects, 131–133
visual effects
    CCParticleSystem, 134
    graphical effects, 134
    Particle Designer, 133
    particle effects, 134

F
Face.com, 264, 280, 298–299
API, 313–314
API Keys tab, 282–283
console output, 285–287
didFinishPickingImage method, 284
FaceDotComFaceDetect method, 283–284
Faces.detect API call, 280–281
REST API, 280
support, 281–282
Facebook application
Auth Delegate methods, 252
dialogDidComplete method, 259
FBDialogDelegate protocol, 259
initial authentication dialog, 252
JSON response, 254–255
login attempts, 253
Request Delegate methods, 251
shareButtonClicked method, 257–258
sharing dialog, 261
updated ARAnnotation, 257
updated didLoad method, 254–256
updated fbDidLogin method, 253
updated initWithCoordinate method, 258–259
URL Schemes, 252
Facebook context
FB App ID, 250
OAuth Dialog, 250
RootViewController, 250
Facebook iOS SDK, 226–227
Facial recognition
ASI-HTTP-Request Library, 314–315
ASI-HTTP-Request Library
downloading, 300
camera
AppDelegate.h After Updates, 304–305
Call Our New Method, 307
front-facing, 305–306
pulling image off the buffer, 308–309
rotateImage Utility Method, 310–312
setImageToView Method, 308
setting up CaptureSession, 306–307
transparent view, 307–308
Updated setImageToView
Method, 312
cocos2D, 299
dealloc method, 304
Face.com, 298–299
face.com API, 313–314
HUD layer construction
FaceDetectionLayer, 325–326
loadCrosshair Method, 323–325
New init Method, 326
Twilio callout, 330–331
Updated
applicationDidFinishLaunching
Method, 327
Updated RootViewController, 327
updateMood Method, 327
initWithHUD Method, 303
JSON-framework, 300
NSTimer creation, 317–320
parsing the output, 320–323
POST Request Method, 316–317
project structure, 300–302
purpose, 297
Static Scene Method, 303
technology, 297–298
Twilio account, 299–300
Updated FaceDetectionLayer.h, 303
Facial recognition techniques, mobile
iOS environment
Face.com, 264, 280
API Keys tab, 282–283
console output, 285–287
didFinishPickingImage method, 284
FaceDotComFaceDetect method, 283–284
Faces.detect API call, 280–281
REST API, 280
support, 281–282
iOS 5 CIDetector Class, 264, 277
advantages, 280
CIDetectorFaceDetect, 278
didFinishPickingImage, 279
iPad simulator, 279
MainViewController.h, 278
open computer vision (OpenCV), 263–264
capturing image for testing, 265–270
Haar cascades, 271–277
iPad, 265
performance measurement
CodeTimestamps.h, 287–288
CodeTimestamps.m, 288–294
console output, 294
didFinishPickingImage method, 294
LogTimestamp macro, 294

Gaming and location-based AR, 4
Geographic Region Monitoring Service
didEnterRegion method, 43–44
didExitRegion method, 44–45
didUpdateToLocation method, 45–46
distance between location updates, 46
FirstViewController.h, 40
FirstViewController.m, 39–41
iOS 5 authorization dialog, 42
preloaded GPS coordinates, Xcode 4.2 beta, 40
startRegionMonitoring method, 40–41
viewDidLoad method, 41
Xcode 4.2 beta location simulator, 43
GitHub, 5–6
Gyroscope, 26–27, 63, 69–72

Hardware components
accelerometer, 26
camera, 16–17
checking for sound, 24
checking for video capability, 24–25
detecting camera, 17–20
detecting location capabilities, 20–21
digital compass, 21–23
gyroscope, 26–27

IBAction method, 72–74
Image compression
JPEG, 110
PNG, 110
iOS 5 authorization dialog, 42
iOS 5 CIDetector Class, 264, 277
advantages, 280
CIDetectorFaceDetect, 278
didFinishPickingImage, 279
iPad simulator, 279
MainViewController.h, 278
iOS Sensors
accelerometer, 64–67
gyroscope, 69–72
IBAction method, 72–74
low-pass filtering, 69
magnetometer
availability, 77
calibration, 78
Open headingViewController.xib in interface builder, 78–81
orientation changes, 81–84
orientation sensors, 81–84
rate of rotation, 74–77
shake detection, 68–69
iPhone simulator, 38

JPEG format, 110
JSON-framework, 300

LoadPhotoPicker method, 109
Location services, iPhone
adding annotations to map
customized, 58
default, 57
displayMap method, 56
MapAnnotation.h changes, 55
MapAnnotation.m changes, 55
viewForAnnotation delegate method, 57
changing map type
CGRectMake method, 52
satellite map type, 54
setupSegmentedControl method, 52–53
UISegmentedControl, 52–53
checking altitude
altitude property, 48
CLLocation method, 48
CLLocationManager method, 47
CLLocationManagerDelegate method, 48
coordinate property, 48
delegate property, 47
desiredAccuracy property, 47
distanceFilter property, 47
GPS reading, 47
location property, 47
locationManager didFailWithError method, 48
locationManager didUpdateToLocation fromLocation method, 48
timestamp property, 48
FirstView controller, 33
Geographic Region Monitoring Service
didEnterRegion method, 43–44
didExitRegion method, 44–45
didUpdateToLocation method, 45–46
distance between location updates, 46
FirstViewController.h, 40
FirstViewController.m, 39–41
iOS 5 authorization dialog, 42
preload GPS coordinates, Xcode 4.2 beta, 40
startRegionMonitoring method, 41
viewDidLoad method, 41
Xcode 4.2 beta location simulator, 43
GitHub, 31
memory management, 33
reverse geocoding, 58
MKReverseGeocoder class, 59
MKReverseGeocoderDelegate methods, 60
toggleToolBarChange method, 59
Xcode console, 60
Significant-Change Location Service
cellular radio, 38
viewDidLoad method, 39
Standard Location Service
CLLocationManagerDelegate method, 36
Core Location Framework, 34–35
FirstViewController header file, 35
iPhone simulator, 38
modal dialog, 37
startStandardUpdates method, 36–37
viewDidLoad method, 37
UILabel, 32–33
UITextView, 32–34
context menu, 34
viewing on map
Apple HQ, 51
centering map and setting displayed region, 50–51
configuring MKMapView, 49
displayMap method, 50
MKCoordinateRegion, 50
MKMapView IBOutlet, 48
MKMapViewDelegate, 48
NSAutoreleasePool class, 51
synthesize and release mapView, 49
thread selector, 50
Low-pass filtering, 69

M
Magnetometer, 21–23, 77
Markers
advantages, 211
characteristics, 211
GitHub repository, 212
high-contrast, unique image, 211
OpenGL ES, 212

N
NSAutoreleasePool class, 51

O
Object recognition. See Facial recognition
Open computer vision (OpenCV)
capturing camera source, 270
cameraButtonClicked method, 268
MainViewController, 266–267
MainViewController.h, 267–268
UIActionSheet Delegate method, 268
UIImagePickerController, 270
UIImagePickerController Delegate method, 269
Haar cascades
building OpenCV static library, 276
calling OpenCV routine, 275–276
createIplImage method, 272
extracting OpenCV routine, 275–276
iPad simulator, 277
limitations, 277
openCVFaceDetect method, 273–274
patching OpenCV routine, 276
private methods, 271
iPad, 265
Open headingViewController.xib in interface builder, 78–81
OpenCV. See Open computer vision (OpenCV)
OpenGL ES. See OpenGL for Embedded Systems (OpenGL ES)
OpenGL features. See Shaders and lighting effects
OpenGL for Embedded Systems (OpenGL ES), 212
framework, 194
Orientation sensors, 81–84

Performance measurement
CodeTimestamps.h, 287–288
CodeTimestamps.m, 288–294
console output, 294
didFinishPickingImage method, 294
LogTimestamp macro, 294
Photo capture
e-mailing image
didFinishSavingWithError, 113
error message, 113
formatted e-mail, 114
PhotoViewController interface, 111–112
sendEmailMessage method, 112–113
saving images in different formats
UIImage Object, 110
UIImageJPEGRepresentation, 111
UIImagePNGRepresentation, 111
using camera
loadPhotoPicker method, 109
methods to save photo image, 109–110
selector method, 110
UIAlertView dialog, 110
UIImagePickerController class, 110
using storyboards
menu, 108
PhotoViewController class creation, 105
tab bar interface, 104
UIButton action, 108
UIImagePickerControllerDelegate, 105
UINavigationControllerDelegate, 105
PNG format, 110
POST Request Method, 314–315

QCAR demo
creating new project, 190
creating new trackable link, 190
creating Xcode project, 192–194
downloading trackable link, 192
EAGLView
configuring video as background image, 204
createFrameBuffer method, 197–198
deleteFrameBuffer method, 198
external methods for
ViewController, 199–200
Handle Status Updates, 200–201
import statements, 194
init methods, 202
initializing OpenGL rendering and render frames, 205–206
initWithCoder, 197
interface, 195
layerClass static method, 196
loading marker/trackable, 201–202
loading test shaders, 204–205
new import statements, 196
new method declarations, 195
presentFramebuffer and layoutSubviews methods, 199
private methods, 196
setFrameBuffer method, 199
starting and stopping camera view, 203
status enumeration, 195
redirecting UIView
demo application, 208
new viewDidLoad, 206–207
new viewWillAppear and viewWillDisappear, 207
updating UIView class to EAGLView, 207
Qualcomm SDK
Dominoes application, demo, 189
downloading QCAR SDK, 187–188
installation directory, 188–189
mixed-reality space, virtual button, 189
QCAR SDK, 188–189
Qualcomm toolkit, 181
QuartzCore.framework, 194

Reference-counted memory system, 51
REST API, 280
Reverse geocoding, 58
MKReverseGeocoder class
coordinate property, 59
delegate property, 59
querying property, 59
start/cancel methods, 59
MKReverseGeocoderDelegate
methods, 60
toggleToolBarChange method, 59
Xcode console, 60

Scenes, Cocos2D, 130
Security.framework, 194
Shaders and lighting effects, 186–187
3D lamp, 186
Shake detection, 68–69
Significant-Change Location Service
cellular radio, 38
viewDidLoad method, 39
Signup button, 4
Sound and active feedback
audio conversion, 89–90
audio data formats, 87–88
audio recorder initialization, 96–100
AVAudioPlayer Class, 93–94
bit rates and quality, 88–89
converting file types, 91–92
file formats, 87–88
media files, 90–91
multiple audio players, 94–96
playing positional sound, 96
playing sound, 92
recording sound, 96
sample rates, 89
System Sound Services, 93
testing, 92
vibration effect, 96
Standard Location Service
CLLocationManagerDelegate
method, 36
Core Location Framework, 34–35
FirstViewController header file, 35
iPhone simulator, 38
modal dialog, 37
startStandardUpdates method,
36–37
viewDidLoad method, 37
Storyboards
menu, 108
PhotoViewController class creation, 105
tab bar interface, 104
UIButton action, 108
UIImagePickerControllerDelegate, 105
UINavigationControllerDelegate, 105
String SDK toolkit, 181
System Sound Services, 93
SystemConfiguration.framework, 194

Tab bar interface, 104
Third party toolkits. See Augmented reality (AR), toolkits
Twilio account, 299–300

UIAccelerometerDelegate Class, 67–68
UIImagePickerController class, 101–102
UIRequiredDeviceCapabilities key, 28–29
UIView
demo application, 208
new viewDidLoad, 206–207
new viewWillAppear and viewWillDisappear, 207
updating UIView class to EAGLView, 207
Use Automatic Reference Counting check box, 103
USPS, eagle icon
creation, 1–2
hologram, 2–3
principles, 2

Vibration, 96
Video capture
frame capture
AVCaptureSession class, 116
captureScreen method, 119–120
EXIF attachments, 121
iPhone storyboard, 116–117
placeholder comment, 119
VideoViewController.h, 117–118
viewDidAppear method, 118
video preview
full-screen video, 115
loadPhotoPicker method, 115
UIImagePickerController object, 115
ViewController, augmented reality
Animation Start and Stop methods, 220
import statements, 219
instance methods, 219
instance variables, 219
Portrait orientation, 221
Projection Matrix, 219
render method, 220
Start and Stop animation, 222
updated viewDidLoad method, 221–222

Xcode 4.2
beta location simulator, 43
creating project, 10–11
linking with GitHub, 8–10
setting, 7–8
Xcode project creation, QCAR demo
AVFoundation.framework, 194
Build Phases tab, 193
Build Settings tab, 193
bundle resources, 194
CoreMedia.framework, 194
CoreVideo.framework, 194
OpenGLES.framework, 194
project settings, 193
QuartzCore.framework, 194
Security.framework, 194
SystemConfiguration.framework, 194