Developing a 3D Model Viewer for iOS using COLLADA and OpenGL ES

Ricardo Rendon Cepeda
Idean / RayWenderlich.com
ricardo@rendoncepeda.com

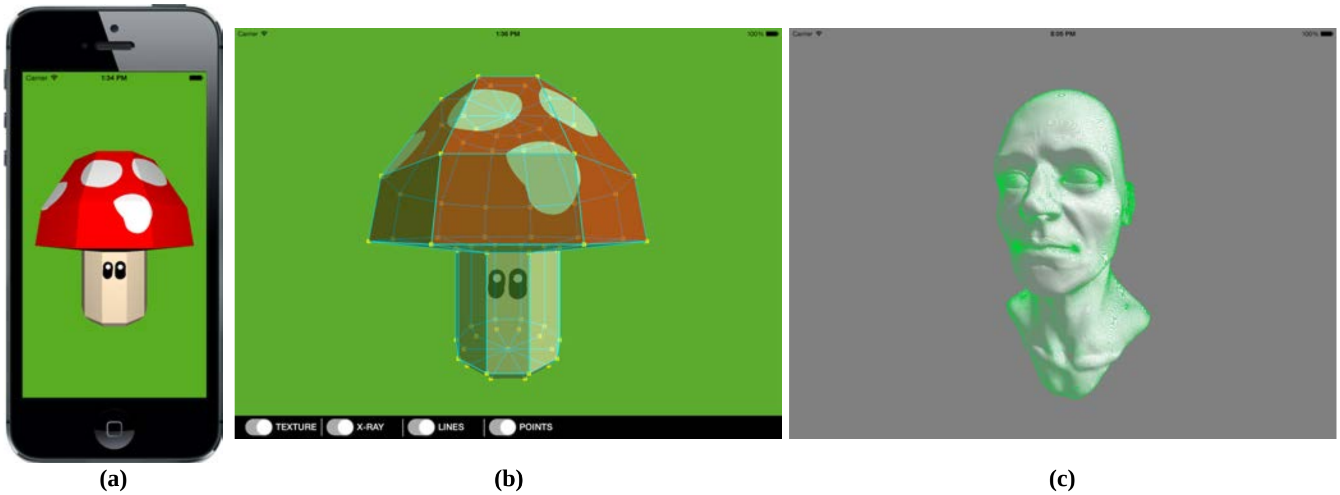

Figure 1: *Screenshots from the completed Xcode project, showcasing 3D COLLADA asset renderings of: (a) a cartoon mushroom in a simple iPhone app, using Apple's GLKBaseEffect; (b) the same cartoon mushroom in an enhanced iPad app, using multiple OpenGL ES shaders, multi-touch gesture recognizers, and basic UI components; (c) a 4.5 million-vertex bust used for system stress testing.*

Abstract

This hands-on, interactive studio course delivers a native iOS app built from scratch, with powerful 3D graphics technologies from the Khronos Group. We explore the latest Xcode IDE and SDK frameworks available to develop the skills and learn the tools needed to make an engaging, mobile, 3D model viewer.

Keywords: Mobile, Studio Course, App Development, Interactive, iOS, COLLADA, OpenGL ES.

1 Intended Audience & Course Prerequisites

This course is aimed at mobile professionals with an interest in 3D graphics and/or graphics professionals with an interest in mobile development. Attendees are assumed to be proficient in an object-oriented programming language and/or familiar with any of the following technologies: iOS, Android, OpenGL ES, OpenGL, COLLADA, XML, Windows Phone 8, Direct3D.

2 Course Requirements & Delivery Format

Each attendee must be equipped with an Apple MacBook running the latest version of Xcode. The code will primarily run on the iOS Simulator built into Xcode, but attendees may optionally choose to deploy to their own iPhone/iPad if they are registered Apple iOS Developers.

A .zip starter pack will be provided (via URL link) with all necessary code templates, pre-built libraries, and digital assets. A full, native iOS app will be developed during the course in an incremental, hands-on session with several "build and run" steps along the way.

3 Course Content

The first half of the course will be a quick introduction to COLLADA and OpenGL ES, noting the data needed from the former to render with the latter. A simple XML parser will be reviewed, integrated, and used to extract COLLADA geometry data - which will then be converted to OpenGL ES arrays. Finally, a quick setup of GLKit and GLKBaseEffect will culminate in a simple rendering of a 3D mushroom model (Fig. 1a).

The second half of the course will be a deeper dive into OpenGL ES, incorporating shaders and GLSL. Three mini-programs will be integrated into the app: Blinn-Phong, Lines, and Points. Finally, the app will spring to life with a series of rich gesture recognizers (pinch, pan, rotation) used to control the 3D scene (scale, translate, rotate).

4 Course Goals

The main takeaway from this studio course is a self-assembled 3D model viewer app that attendees can use to showcase their own models (Fig. 1b). Furthermore, attendees will also be encouraged to further customize their app by adding their own UI elements, shaders, backend integration, or other exciting features!

Links

Project Repository:
https://github.com/ricardo-rendoncepeda/RRCModelViewer

SIGGRAPH 2014, August 10 – 14, 2014, Vancouver, British Columbia, Canada.

ACM 978-1-4503-2977-4/14/08