

헬로, 안드로이드

14주차 - OpenGL의 3D 그래픽 (1)

강대기

동서대학교 컴퓨터정보공학부

학습 목표

- 3 차원 그래픽의 기본 개념에 대해 이해한다.
- OpenGL의 기본 개념에 대해 이해한다.
- 간단한 OpenGL 프로그램을 만들 수 있다.

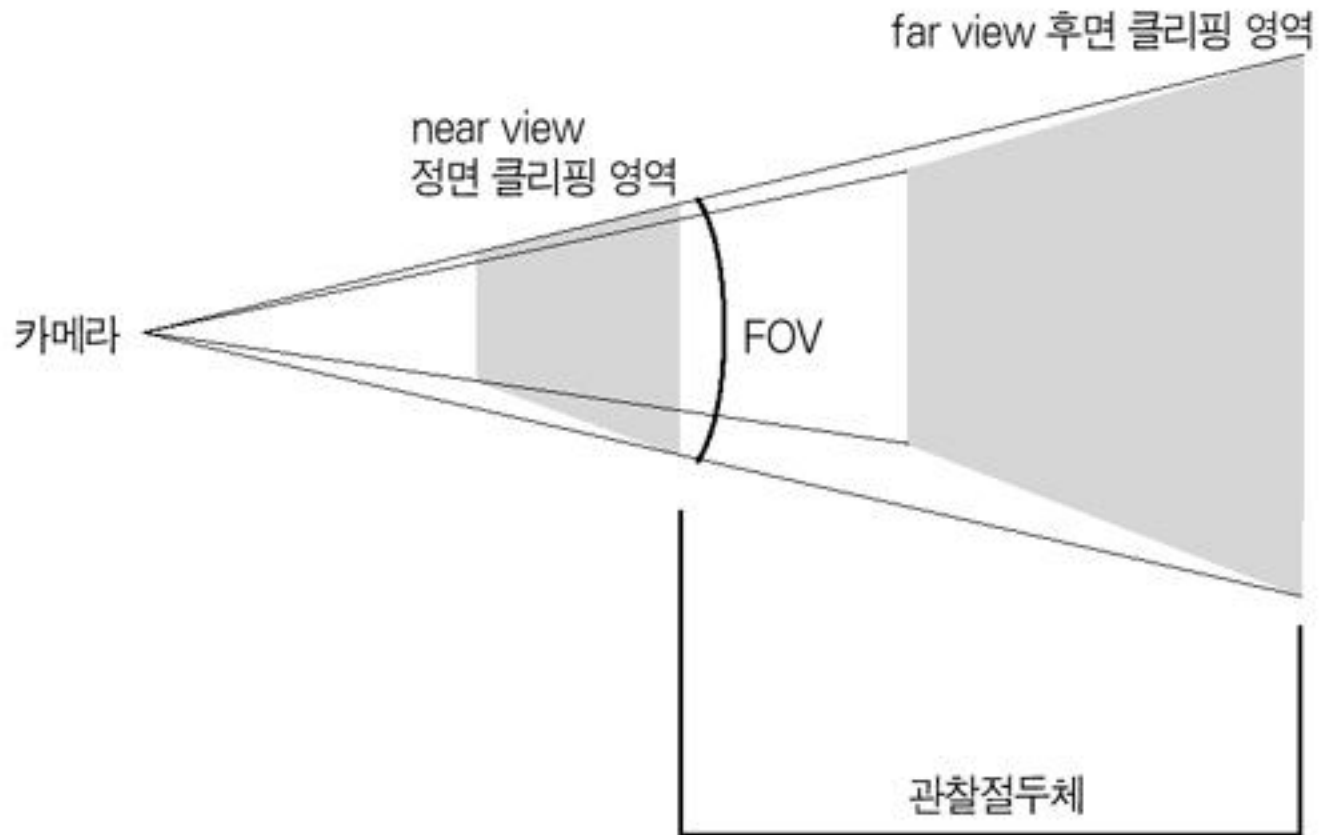
차례

- 3D 그래픽 이해하기
- OpenGL 소개하기
- OpenGL 프로그램 만들기
- 스레드 관리하기
- 배경 화면 변경 및 사용자 이미지로 변경
- 요약
- 퀴즈
- 연습문제

3D 그래픽 이해하기

- 세상은 3차원이나, 책이나 그림, 컴퓨터 모니터를 볼 때에는 2차원을 보고 있음
- 실내에서 창 밖을 바라보면, 그게 바로 컴퓨터 모니터를 통해 보는 것과 비슷함 – 고정된 위치에서 한 쪽 눈으로만 보면, 고화질의 사진으로 바꾸어 놓아도 눈치챌 수 없음
 - 창문 – 뷰포트 (viewport)
 - 눈과 창문 사이의 거리 – 시야 (FOV; field of view)
 - 눈과 창문의 네 귀퉁이를 잇는 선 – 시야 절두체 또는 관찰 절두체 (view frustum)

3차원 장면 보기



OpenGL 소개하기

- 1992년 실리콘 그래픽스사에서 개발
- 개발사가 제조사에 상관없이 하드웨어를 사용할 수 있도록, 통일된 인터페이스 제공
- 핵심 아이디어는 개발자가 하드웨어의 디테일보다는 뷰포트, 조명 등의 개념에만 집중하게 해 줌
- 존 카맥(John Carmack) – Microsoft의 Direct3D에 맞서, Doom과 Quake 등을 OpenGL로 개발하는 것을 고수함
- OpenGL 자체는 워크스테이션 용이어서 임베디드 시스템 용으로는 덩치가 큼 – 안드로이드에서는 임베디드 시스템을 위한 OpenGL ES를 구현
- OpenGL ES는 여러 회사들의 컨소시엄인 Khronos Group에서 만듦
- OpenGL ES의 자바 언어 바인딩은 Java Specification Request (JSR) 239에서 정의됨

OpenGL 프로그램 만들기

- 프로젝트 생성
 - OpenGL / org.example.opengl / OpenGL / OpenGL
- 생성된 OpenGL 액티비티가 GLView 라는 View를 참조하도록 수정함 - res/layout/main.xml 리소스는 삭제
- GLView.java 의 새로운 클래스들 (교재와 다름)
 - GLSurfaceView – SurfaceView의 하위 클래스로 내부적으로 GL 요청을 처리하기 위한 기능을 담당함.
 - GLSurfaceView.Renderer – GLSurfaceView가 가지고 있는 Renderer 인터페이스로, OpenGL이 호출하면 프레임에 그림을 그림
 - EGLConfig – Embedded Graphic Library (EGL) 의 사양(configuration) 을 저장함
 - GL10 – OpenGL(R) ES 1.0 코어의 자바 프로그래밍 언어의 바인딩을 가지는 인터페이스
 - GLU – GL Utilities (GL 유틸리티 함수들)
- 교재의 클래스들 소개
 - SurfaceView – 3차원 그래픽에 사용되는 특별한 종류의 뷰. OpenGL의 모든 뷰로 확장 가능함
 - Surface – Canvas처럼 그리기에 사용되지만, 3D 하드웨어로 구현된다는 차이가 있음.
 - SurfaceHolder – 이 클래스의 인스턴스는 자신이 갖고 있는 Surface가 사라지더라도 계속 유지됨
 - SurfaceHolder.Callback – SurfaceView는 이 인터페이스를 구현하는 데, 뷰의 Surface가 생성되거나 없어지거나, 크기가 변하면, OpenGL이 이를 알림

GLView.java

```
package org.example.opengl;

import android.content.Context;
import android.opengl.GLSurfaceView;

class GLView extends GLSurfaceView {
    private final GLRenderer renderer;

    GLView(Context context) {
        super(context);

        // Uncomment this to turn on error-checking and logging
        //setDebugFlags(DEBUG_CHECK_GL_ERROR |
        DEBUG_LOG_GL_CALLS);

        renderer = new GLRenderer(context);
        setRenderer(renderer);
    }
}
```


GLRendererer.java (대략적인 구성)

```
package org.example.opengl;

import javax.microedition.khronos.egl.EGLConfig;
import javax.microedition.khronos.opengles.GL10;

import android.content.Context;
import android.opengl.GLSurfaceView;
import android.opengl.GLU;
import android.util.Log;

class GLRendererer implements GLSurfaceView.Renderer {
    private final Context context;
    GLRendererer(Context context) {
        this.context = context;
    }
    public void onSurfaceCreated(GL10 gl, EGLConfig config) { ... }
    public void onSurfaceChanged(GL10 gl, int width, int height) { ... }
    public void onDrawFrame(GL10 gl) { ... }
}
```

쓰레드 관리하기

- 안드로이드 2D 라이브러리는 화면의 일부를 다시 그려야 할 때, 뷰의 `onDraw()`를 호출하지만, OpenGL은 그리는데 사용될 렌더러의 `onDrawFrame()`을 호출함 (1초에 프레임을 호출하는 수 - Frames Per Second)
- `GLSurfaceView`는
 - 프레임(frame)에 그림을 그릴 `Renderer` 인터페이스와
 - `Renderer`를 주기적으로 호출할 `GLThread` 클래스,
 - EGL 설정을 담당할 `EGLHelper` 클래스를 가지고 있음
- `GLSurfaceView`를 확장한 `GLView`에서 `GLRenderer` 설정
- `GLRenderer`에서
 - `surface`가 생성되면 (`onSurfaceCreated()`) 프레임을 그리기 위한 설정을 완료하고,
 - `surface`가 바뀌면(`onSurfaceChanged()`) 그에 따른 처리를 하며, (예제에서는 시야 절두체 설정)
 - 프레임을 그려달라는 요청을 받으면(`onDrawFrame()`) 프레임을 그림

onSurfaceCreated()

- 광원이 되는 조명의 특성을 설정함
- 물체의 재료에 따라 다른 질감을 가지므로 이를 설정함
- OpenGL 옵션 설정
- 깊이 테스트 및 알파 블렌딩 설정
 - 알파 블렌딩 - 두 색을 섞을 때, 투명도를 나타내는 알파 채널을 이용하여 섞는 것 - 주로 유리를 통해 본 이미지나 투명한 물 속의 물체를 나타낼 때 사용함
- 질감을 위한 텍스춰 설정
- 필요하면 비트맵 파일을 읽어들이어 텍스춰로 입힘

onSurfaceChanged()

- 시야 절두체 (view frustum) 설정
- 픽셀이 정사각형이 아닐 수 있으므로, 화면 가로와 세로의 ratio 설정
- 투영법 (projection) 설정
 - 많이 쓰이는 것은 원근 투영법
 - 평행 투영법도 있음

onDrawFrame()

- 화면을 검게 청소함
- 이미 읽어들이는 모델을 위치시킬 장소 설정
- 시간에 맞추어 모델을 회전시킬 것이므로, 각도 설정
- 모델을 그림 (예제에서는 정육각형; cube)
- 그려진 프레임의 개수를 계속 추적하여, Frames Per Second (fps) 값 계산함

OpenGL ES

- OpenGL ES 1.0은 통합 OpenGL 1.3, ES 1.1은 통합 OpenGL 1.5에 기초함
- JSR 239 는 원본 1.0과 유지 릴리즈 버전 1.0.1 두 가지 버전이 있음

GLRendererer.java (대략적인 구성)

```
package org.example.opengl;

import javax.microedition.khronos.egl.EGLConfig;
import javax.microedition.khronos.opengles.GL10;

import android.content.Context;
import android.opengl.GLSurfaceView;
import android.opengl.GLU;
import android.util.Log;

class GLRendererer implements GLSurfaceView.Renderer {
    private final Context context;
    GLRendererer(Context context) {
        this.context = context;
    }
    public void onSurfaceCreated(GL10 gl, EGLConfig config) { ... }
    public void onSurfaceChanged(GL10 gl, int width, int height) { ... }
    public void onDrawFrame(GL10 gl) { ... }
}
```

배경 화면 변경 및 사용자 이미지 삽입

- 새로 AVD를 생성하되, SD 카드를 충분히 크게 잡아 줌 (64M 정도?)
- Camera를 실행하여 사진 찍음
- Home의 Menu 에서 Wallpaper 선택 후 Pictures 선택하면 찍힌 사진이 보임
- adb 프로그램으로 SD 카드가 마운트된 디렉토리에서 찍힌 사진을 복사함
 - `C:\Android\android-sdk-windows-1.5_r3\tools>adb pull "/sdcard/DCIM/Camera/2009-10-07 16.12.35.jpg" camera_image.jpg`
- 올리고 싶은 사진을, 복사된 카메라 이미지 (camera_image.jpg)와 같은 크기로 만든 뒤, adb로 올림 (213 x 350 x 24) – GIMP 소프트웨어 사용, 파일명은 소문자로 함
- 에뮬레이터를 재시동

요약

- 3 차원 그래픽의 기본 개념을 다루어 보았다.
- OpenGL의 기본 개념에 대해 공부해 보았다.
- 간단한 OpenGL 프로그램을 만들어 보았다.

퀴즈

- Canvas와 Surface의 차이는 무엇인가?
- 뷰포트 (viewport) 란 무엇인가?
- 시야 (FOV; field of view) 란 무엇인가?
- 시야 절두체 (view frustum) 란 무엇인가?
- OpenGL ES 란 무엇인가?
- OpenGL ES 의 자바 언어 바인딩은 무엇으로 정의되어 있는가?
- GLSurfaceView 클래스에 대해 설명하라.
- GLSurfaceView 의 GLRenderer 는 무슨 일을 하는가?
- 알파 블렌딩이란 무엇인가?
- 원근 투영법이란 무엇인가?
- FPS는 무엇인가?

연습문제

- 정육면체 대신 구체 (ball), 또는 정사면체, 원뿔, 원기둥 등을 모델로 설정해서 프로그램을 작성하라.
- Spinner 위젯을 이용하여 정육면체, 구체, 정사면체, 원기둥, 원뿔 중 모델을 선택할 수 있게 해 보자. 이를 위해서 XML 파일로 이루어진 메인 뷰 안에 사용자의 Custom 뷰를 저장하는 방법을 사용해 보자.
- 예제 프로그램은 GLSurfaceView 에서 GLRenderer를 사용하고 있다. 교재에 나와있는 것처럼 쓰레드를 이용하는 방법으로 예제 프로그램을 재구성해 보라.