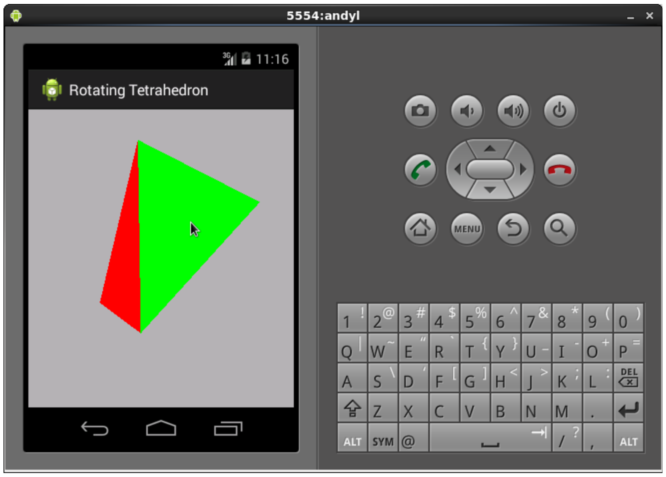
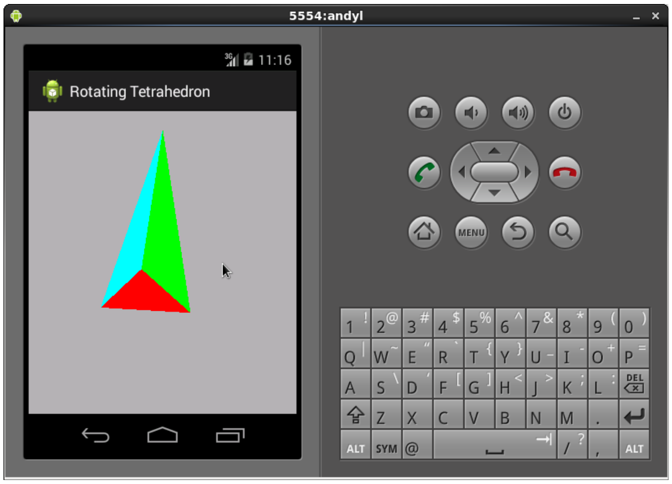
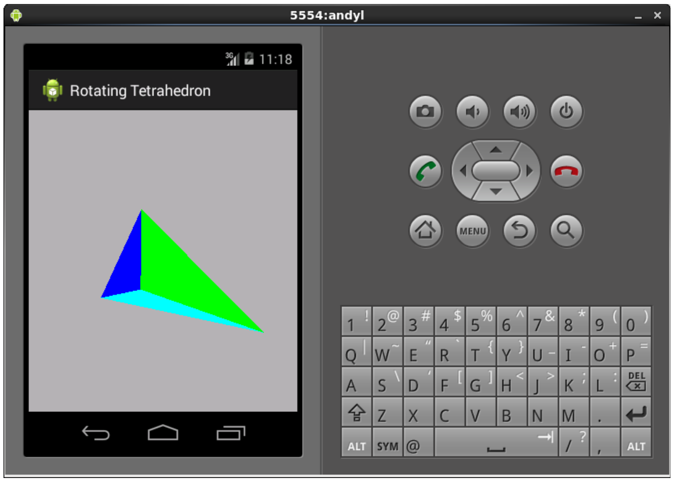
Yazhuo Liu
Lab 4

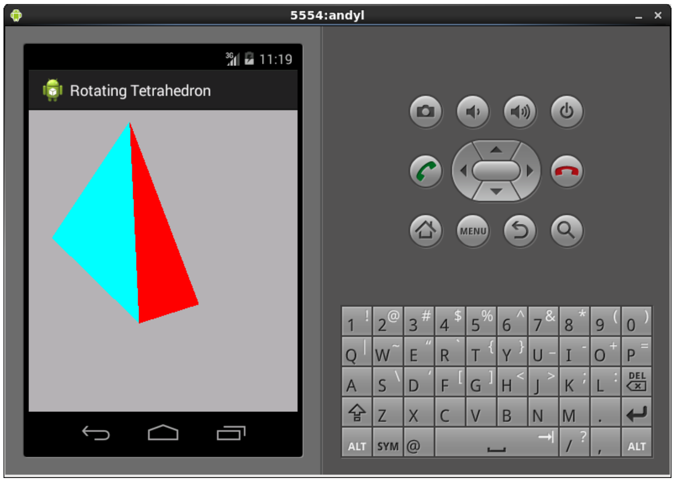
Write an Android graphics program using OpenGL ES 1X .that renders a colored tetrahedron that allows the user to rotate it by dragging the mouse.





Then modify your program so that the object can rotate along the x, y or z axis automatically.





Code:
//MainActivity.java
...

```
 class TetraSurfaceView extends GLSurfaceView {
   private final float TOUCH_SCALE_FACTOR = 180.0f / 320;
   private TetraRenderer renderer;
   private float previousX;
   private float previousY;

    public TetraSurfaceView(Context context){
      super(context);
      renderer = new TetraRenderer();
      // Set the Renderer for drawing on the GLSurfaceView
      setRenderer(new TetraRenderer());
      // Render the view only when there is a change
```

```
            //setRenderMode(GLSurfaceView.RENDERMODE_WHEN_DIRTY);
        }

        @Override
        public boolean onTouchEvent(MotionEvent e) {
            // MotionEvent reports input details from the touch screen
            // and other input controls. Here, we are only interested
            // in events where the touch position has changed.
            float x = e.getX();
            float y = e.getY();
            switch (e.getAction()) {
                  case MotionEvent.ACTION_MOVE:
                        float dx = x - previousX;
                        float dy = y - previousY;
            // reverse direction of rotation above the mid-line
                        if (y > getHeight() / 2)
                              dx = dx * -1;
            // reverse direction of rotation to left of the mid-line
                        if (x < getWidth() / 2)
                              dy = dy * -1;
                        renderer.angle += (dx + dy) * TOUCH_SCALE_FACTOR;
                        requestRender();
            }
            previousX = x;
            previousY = y;
            return true;
        }
    }
```

...

//TetraRenderer.java

...

```
    public void onDrawFrame(GL10 gl) {
        // Redraw background color
        gl.glClear(GL10.GL_COLOR_BUFFER_BIT | GL10.GL_DEPTH_BUFFER_BIT);
        gl.glMatrixMode(GL10.GL_MODELVIEW);
        gl.glLoadIdentity();
        //gl.glRotatef(-50, 1, 0, 0);

        GLU.gluLookAt(gl, -4, -4, 5, 0.2f, 0.2f, 0f, 0f, 1.0f, 0.0f);
        SystemClock.sleep ( 300 );
        angle += 10;
        //rotate triangle
        gl.glRotatef(angle, 1, 1, 1);
        //magnify triangle
        gl.glScalef ( 1, 0.8f, 0.8f);
```

...

Report:

I believe I have completed all parts of this lab successfully.