

SPRAWOZDANIE

Zajęcia: Grafika komputerowa

Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium 7

Data: 19.04.2022

Temat: "Tekstury w OpenGL"

Michał Wielopolski
Informatyka I stopień,
stacjonarne,
4 semestr,
Gr. 4

1. Polecenie:

Celem jest teksturowanie piramidy z użyciem dwóch sposobów ładowania tekstur: użycie tekstury z buforu kolorów (rysowanie w Panel); ładowanie tekstury z pliku (trzy pliki przykładowe do pobrania). Należy opracować metody `textureFromPainting()` oraz `textureFromResource()` klasy `Lab7`.

2. Wykorzystane komendy:

a) kod źródłowy

```
private Texture textureFromPainting() {
    // TODO: write this method

    GLContext context = displayGL.getContext(); // OpenGL context for the
display panel.

    if ( ! context.isCurrent() ) {
        // Make the context current on the current thread.
        context.makeCurrent();
    }
    GL2 gl2 = context.getGL().getGL2();

    BufferedImage img = paintPanel.copyOSC();
    Texture texture;
    ImageUtil.flipImageVertically( img );
    texture = AWTTextureIO.newTexture(displayGL.getGLProfile(),
img, true);
    texture.setTexParameteri(gl2, GL2.GL_TEXTURE_WRAP_S,
GL2.GL_REPEAT);
    texture.setTexParameteri(gl2, GL2.GL_TEXTURE_WRAP_T,
GL2.GL_REPEAT);
    return texture;
}
```

b) kod źródłowy

```
private void paintingFromOpenGL() {
    GLContext context = displayGL.getContext(); // OpenGL context for the
display panel.
    boolean needsRelease = false; // Will be set to true if context
needs to be made current.
    if ( ! context.isCurrent() ) {
        // Make the context current on the current thread.
        context.makeCurrent();
        needsRelease = true;
    }
    GL2 gl2 = context.getGL().getGL2();
    AWTGLReadBufferUtil readBuf = new
AWTGLReadBufferUtil(displayGL.getGLProfile(), false);
}
```

```

        BufferedImage img = readBuf.readPixelsToBufferedImage(gl2, true); //
Get display content as image.
        if (needsRelease) {
            context.release();
        }
        paintPanel.installImage(img); // copy the image into the PaintPanel.
    }

```

c)

```

public void display(GLAutoDrawable drawable) {

    GL2 gl2 = drawable.getGL().getGL2(); // The object that contains all
the OpenGL methods.

    gl2.glClear( GL2.GL_COLOR_BUFFER_BIT | GL2.GL_DEPTH_BUFFER_BIT );

    camera.apply(gl2); // Sets projection and view transformations.

    // TODO: apply currentTexture (or turn off texturing if it is null)

    if(currentTexture !=null) {
        currentTexture.bind(gl2);
        gl2.glEnable( GL2.GL_TEXTURE_2D );
        currentTexture.bind(gl2);
    }else
    {
        gl2.glDisable( GL2.GL_TEXTURE_2D );
    }

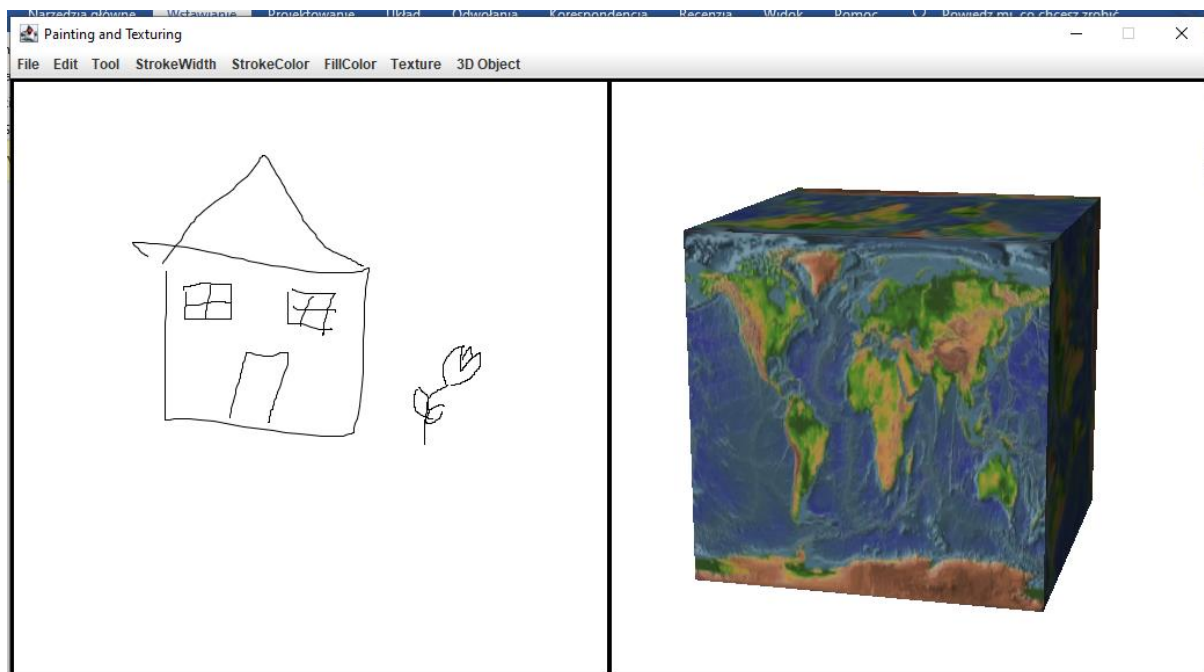
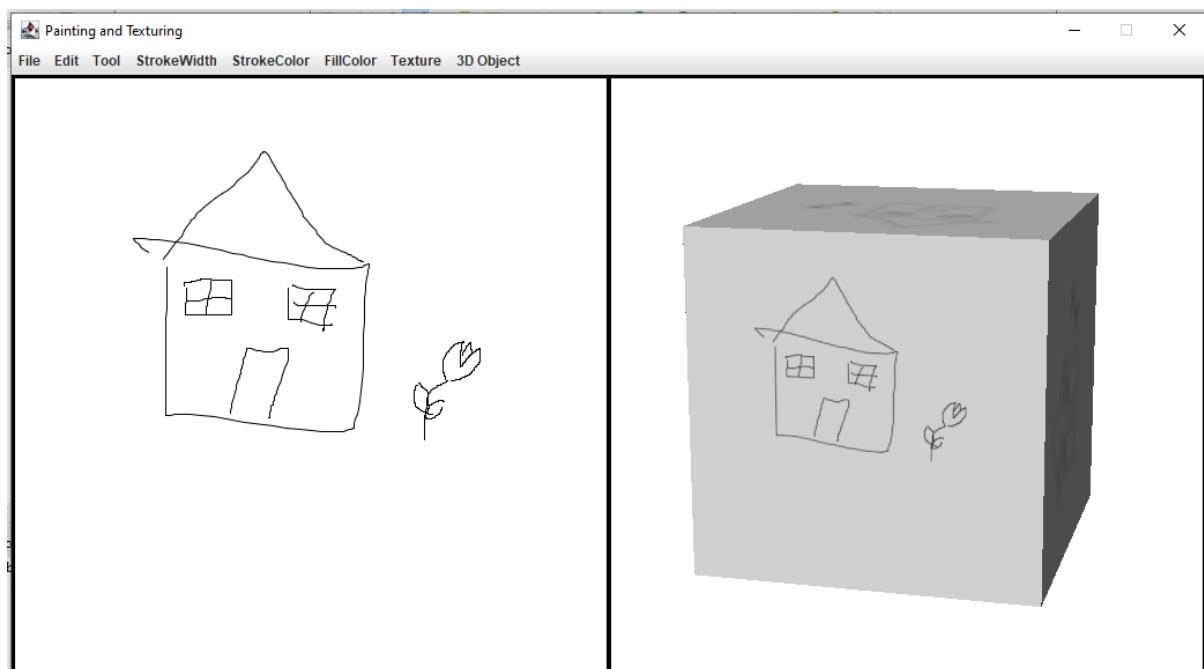
    drawCurrentShape(gl2);

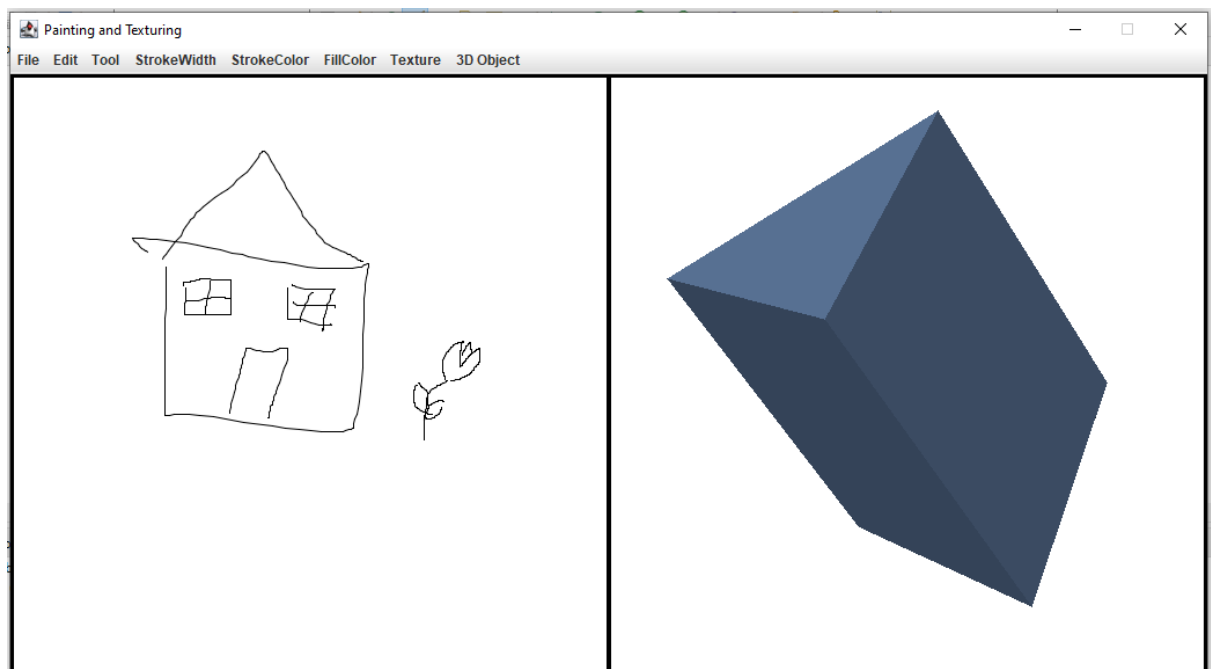
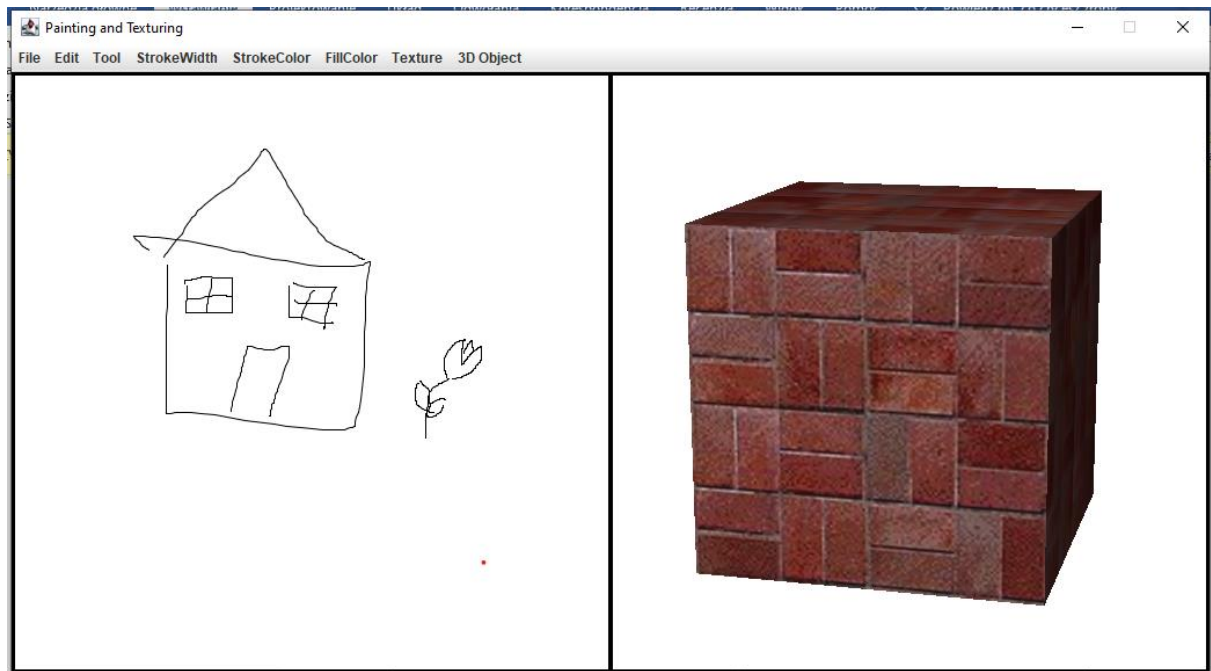
} // end display()

```

<https://github.com/wielopolski/GrafikKomputerowa>

4. Wynik działania:





5. Wnioski:

Za pomocą Javy i OpenGL, możemy dokonywać zmiany materiałów, modyfikacji tekstur i tworzyć prymitywne obrazki.