

Sprawozdanie nr 5, Teksturowanie

Łukasz Szumilas, Grupa E02-81o

Zajęcia: 19.12.2018 (odrobione 28 stycznia)

1 Omówienie tematu

Celem ćwiczenia było pokazanie podstawowych technik teksturowania powierzchni obiektów z wykorzystaniem mechanizmów biblioteki OpenGL z rozszerzeniem GLUT. Zostało przedstawione krok po kroku odpowiednie wczytywanie tekstur obrazka o odpowiednim formacie. Teksturowanie odbyło się na przykładach trójkąta, wielościanu i bardziej skomplikowanego modelu w postaci siatki trójkątów (jajka).

Na rozgrzewkę, przy użyciu programów z poprzednich laboratoriów, należało przedstawić biały trójkąt oświetlony białym światłem. Wykorzystany był program napisany w ćwiczeniu 4 pozwalający obserwować figurę z punktu na powierzchni sfery. Dla ilustracji oświetlenia pomocne były instrukcje z ćwiczenia 5. Efekty widać na **rysunku nr 1**.

Po wykonaniu tej części należało wstawić teksturę na oświetlony trójkąt przy użyciu gotowej funkcji *GLbyte *LoadTGALmage*, należało także dopisać kilka linii kodu w funkcji *MyInit()*, zawierającego podstawowe funkcje definiujące właściwości teksturowania. Ważne było wprowadzenie odpowiednich współrzędnych wzorca tekstury definiującej fragment obrazu, jaki ma być naniesiony na powierzchnię trójkąta. Efekt tego przedsięwzięcia widnieje na **rysunku nr 2**. Dzięki informacjom wyniesionym z laboratorium 'Interakcja z użytkownikiem' trójkąt można obracać, dzięki temu lepiej można zbadać wczytaną teksturę (**rysunek nr 3**).

Następnym etapem było obłożenie teksturą figury 3D, ostrosłupa. W funkcji *RenderScene()*, dzięki odpowiednim połączeniom zdefiniowanych punktów w przestrzeni trójwymiarowej można było połączyć cztery pochylone trójkąty z podstawą kwadratu, łącząc w ten sposób ostrosłup. Z racji tego, że teksturowanie było prowadzone tylko po jednej stronie każdej ściany, trzeba było mieć na uwadze odpowiednią kolejność łączonych punktów. Najważniejsze rzeczy zostały zawarte **kodzie nr 1 i rysunku nr 4**.

Nakładanie tekstury na figurę jajka było o tyle trudniejsze, że figura ta nie ma płaskiej powierzchni i bez odpowiedniego rozplanowania punktów tekstury w *glTexCoord2f(..)* obraz może być całkowicie nieczytelny. Tekstura została pokryta taką samą siatką trójkątów jaką naniesiono na dziedzinę parametryczną, z zachowaniem odpowiedniej kolejności. Tekstury w formacie *tga* były przygotowane specjalnie do tego laboratorium, ale nic nie stoi na przeszkodzie w zdefiniowaniu własnej, przy konwersji pliku w odpowiednio dedykowanym do tego programie. W tym przypadku pomocna była strona <https://www.online-convert.com>. Efekt: **rysunek nr 5 i 6**.

2 Omówienie kodu

Kod 1, fragment funkcji *RenderScene()*.

```
glColor3f(1.0f, 1.0f, 1.0f); // kolor biały

glBegin(GL_QUADS);

glTexCoord2f(1.0f, 0.0f);
```

```

glVertex3f(5.0f, 0.0f, 5.0f);

glTexCoord2f(0.0f, 0.0f);
glVertex3f(-5.0f, 0.0f, 5.0f);

glTexCoord2f(0.0f, 1.0f);
glVertex3f(-5.0f, 0.0f, -5.0f);

glTexCoord2f(1.0f, 1.0f);
glVertex3f(5.0f, 0.0f, -5.0f);

glEnd();

glBegin(GL_TRIANGLES);

glTexCoord2f(0.5f, 2.0f);
glVertex3f(0.0f, 10.0f, 0.0f);

glTexCoord2f(0.0f, 0.0f);
glVertex3f(5.0f, 0.0f, 5.0f);

glTexCoord2f(1.0f, 0.0f);
glVertex3f(5.0f, 0.0f, -5.0f);

glBegin(GL_TRIANGLES);

glTexCoord2f(1.0f, 0.0f);
glVertex3f(-5.0f, 0.0f, 5.0f);

glTexCoord2f(0.5f, 2.0f);
glVertex3f(0.0f, 10.0f, 0.0f);

glTexCoord2f(0.0f, 0.0f);
glVertex3f(-5.0f, 0.0f, -5.0f);

glEnd();

glBegin(GL_TRIANGLES);

glTexCoord2f(1.0f, 0.0f);
glVertex3f(-5.0f, 0.0f, 5.0f);

glTexCoord2f(0.0f, 0.0f);
glVertex3f(5.0f, 0.0f, 5.0f);

glTexCoord2f(0.5f, 2.0f);
glVertex3f(0.0f, 10.0f, 0.0f);

glEnd();

glBegin(GL_TRIANGLES);

glTexCoord2f(1.0f, 0.0f);

```

```

glVertex3f(-5.0f, 0.0f, -5.0f);

glTexCoord2f(0.5f, 2.0f);
glVertex3f(0.0f, 10.0f, 0.0f);

glTexCoord2f(0.0f, 0.0f);
glVertex3f(5.0f, 0.0f, -5.0f);

glEnd();

glFlush();
glutSwapBuffers();

```

Kod 2, nakładanie tekstury na jajko.

```

//i, j – kolejne punkty w przestrzeni dwuwymiarowej (0:1)
//dla których obliczane były wartości dla jajka
//N – liczba podziału tej przestrzeni, w tym przypadku N = 100

//pominięte fragmenty dot. oświetlenia, obrotu i
//obliczania współrzędnych jajka, wykorzystane z poprzednich lab
//i zawarte w poprzednich sprawozdaniach
    if (i > 0 && j > 0) {
        glBegin(GL_TRIANGLES);
        glTexCoord2f(i / N, j / N);
        glVertex3fv(tabXYZ[i][j]);

        glTexCoord2f((i + 1) / N, j / N);
        glVertex3fv(tabXYZ[(i + 1)][j]);

        glTexCoord2f(i / N, (j + 1) / N);
        glVertex3fv(tabXYZ[i][j + 1]);

        glEnd();

        glBegin(GL_TRIANGLES);
        glTexCoord2f((i + 1) / N, j / N);
        glVertex3fv(tabXYZ[(i + 1)][j]);

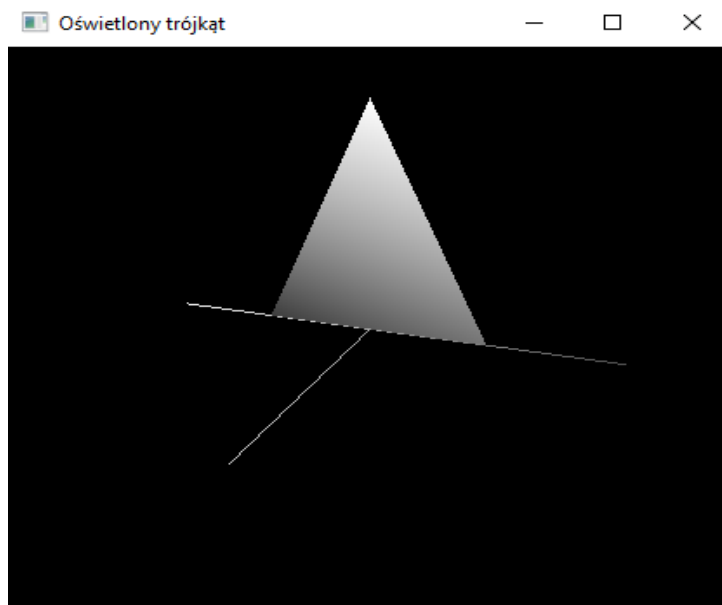
        glTexCoord2f((i + 1) / N, (j + 1) / N);
        glVertex3fv(tabXYZ[(i + 1)][j + 1]);

        glTexCoord2f(i / N, (j + 1) / N);
        glVertex3fv(tabXYZ[i][j + 1]);

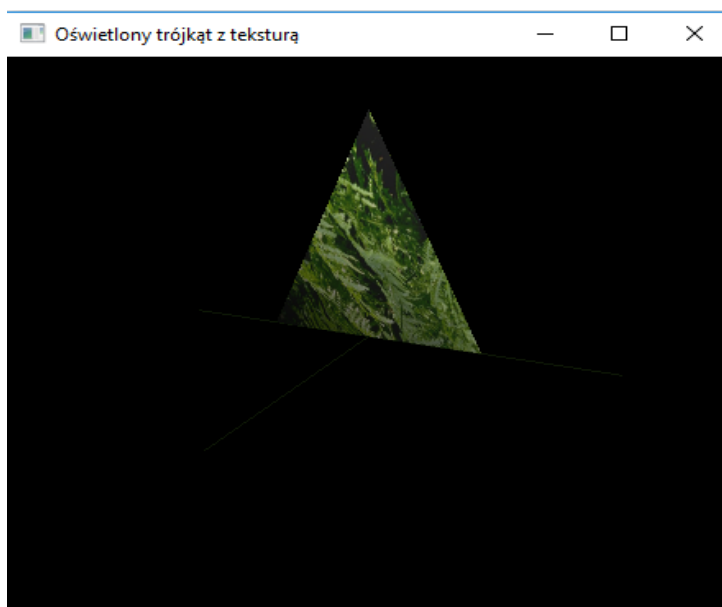
        glEnd();
    }
}

```

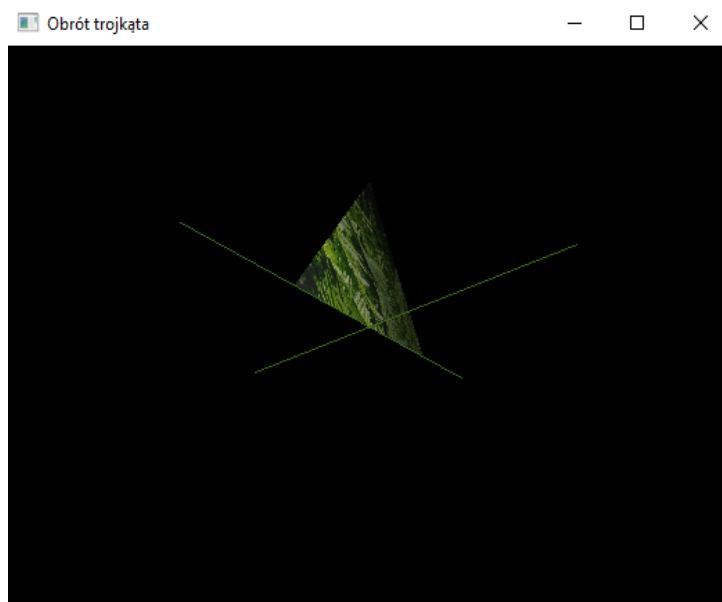
3 Rezultat prac



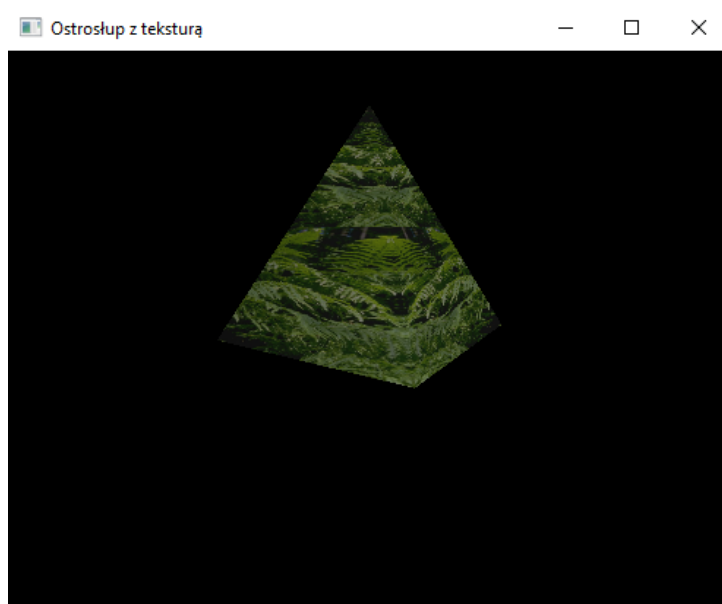
Rysunek 1: Oświetlony trójkąt. Białe światło, biały kolor.



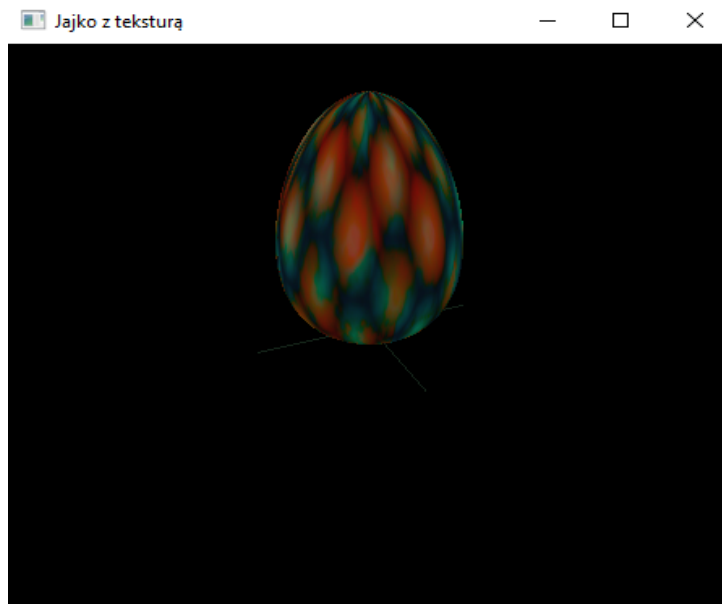
Rysunek 2: Nałożona tekstura na trójkąt.



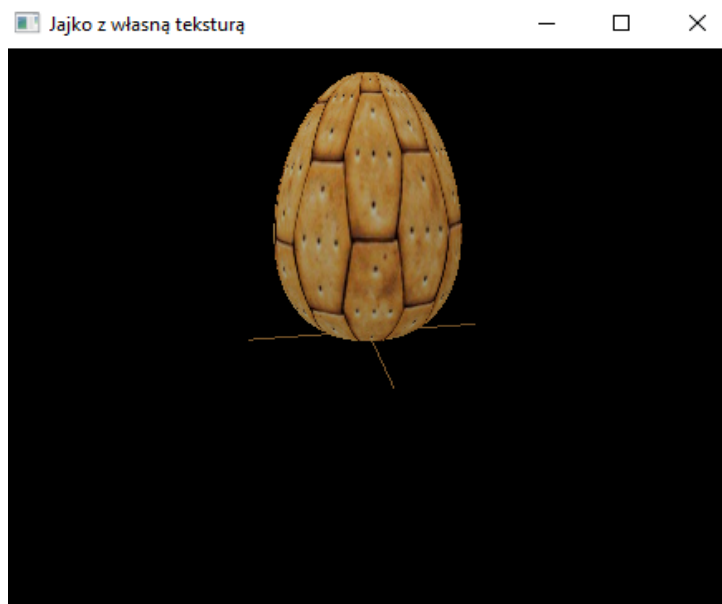
Rysunek 3: Obrócony trójkąt.



Rysunek 4: Ostrosłup z nałożoną teksturą.



Rysunek 5: Jajko z teksturą.



Rysunek 6: Jajko z własną teksturą.