

Instrukcja laboratoryjna 13	Grafika komputerowa 3D
	Temat: Tekstutowanie
	Przygotował: dr inż. Grzegorz Łukawski, mgr inż. Maciej Lasota, mgr inż. Tomasz Michno

1 Wstęp teoretyczny

Tekstutowanie jest jednym z najważniejszych elementów grafiki 3D, ponieważ pozwala znacznie zwiększyć realistyczność tworzonej sceny. Dodatkowo umożliwia pokazanie szczegółów bez konieczności tworzenia osobnych prymitywów, co znacznie przyspiesza rendering. Istnieją 3 rodzaje tekstur: 1D, 2D i 3D. Każdą z nich włącza się za pomocą glEnable, natomiast wyłącza z użyciem glDisable.

Tekstury 1D (jednowymiarowe) są to tekstury, które posiadają tylko jeden wymiar – szerokość lub wysokość (drugi wymiar ma zawsze rozmiar 1 piksela). Typ w OpenGL'u to: GL_TEXTURE_1D.

Tekstury 2D (dwuwymiarowe) są najczęściej spotykanymi rodzajami tekstur. Posiadają wysokość oraz szerokość, tak jak zwykłe obrazki, jednak ich rozmiary muszą być potęgami liczby 2 (np. 128x128, 128x512 itp.). Typ w OpenGL'u to: GL_TEXTURE_2D.

Tekstury 3D¹ (trójwymiarowe, wolumetryczne) są teksturami, które pozwalają na „wypełnienie” obiektu również w środku. Dzięki temu możliwe jest odwzorowanie np. wnętrza Ziemi. Typ w OpenGL'u to: GL_TEXTURE_3D.

Maksymalny rozmiar tekstury zależy od posiadanej karty graficznej, jednak rozmiary do 512x512 powinny być obsługiwane bez problemów nawet przez kilkuletnie karty graficzne.

1.1 Tworzenie tekstur

Do utworzenia tekstury w bibliotece OpenGL służy funkcja:

```
void glGenTextures(GLsizei n, GLuint *textures)
```

gdzie:

n – liczba generowanych tekstur

textures - wskaźnik na element tablicy typu GLuint, w którym będzie przechowywana

¹ Więcej o teksturach 3D i używaniu ich w OpenGL'u można znaleźć pod adresem: http://content.gpwiki.org/index.php/OpenGL:Tutorials:3D_Textures

wygenerowana tekstura.

Następnie należy powiązać teksturę z jej nazwą, za pomocą:

```
void glBindTexture(GLenum target, GLuint texture)
```

Pierwszym parametrem jest rodzaj tekstury (GL_TEXTURE_1D, GL_TEXTURE_2D lub GL_TEXTURE_3D). Drugi parametr jest tablicą typu GLuint, w której przechowywana będzie tekstura. Aby zwolnić obiekty tekstur wykorzystujemy funkcję glDeleteTextures:

```
void glDeleteTextures(GLsizei n, const GLuint *textures)
```

1.2 Tworzenie obrazów tekstury

Gdy mamy już wygenerowaną oraz powiązaną teksturę za pomocą funkcji glGenTextures oraz glBindTexture możemy jej przypisać konkretne dane obrazowe. W OpenGL do tworzenia obrazów tekstur **jednowymiarowych** służy funkcja glTexImage1D.

Funkcja ta wymaga przekazania ośmiu parametrów.

```
void glTexImage1D(GLenum target, GLint level, GLint components,  
                 GLsizei width, GLint border, GLenum format, GLenum type,  
                 const GLvoid *pixels)
```

Argument target określa, która tekstura powinna zostać zdefiniowana, musi nim być GL_TEXTURE_1D. Argument level wskazuje poziom szczegółów obrazu tekstury i zwykle wynosi 0 (inne wartości używane są przy teksturach typu mipmap). Argument components określa ilość wartości koloru użytych dla każdego piksela. Argument width i border określa rozmiar obrazu tekstury. Wartość border określa ilość pikseli ramki może mieć wartość 0, 1 lub 2. Parametr width określa szerokość głównego obrazu tekstury (bez pikseli ramki) i musi być potęgą liczby 2. Argument format wskazuje rodzaj wartości koloru w teksturze (GL_COLOR_INDEX, GL_LUMINANCE, GL_RGB, GL_RGBA). Parametr type określa typ danych dla wartości pikseli. Natomiast ostatni parametr pixels stanowi dane pikseli (które należy np. wczytać z pliku).

W celu zdefiniowania **dwuwymiarowej** tekstury należy wywołać funkcję glTexImage2D. Funkcja ta przyjmuje identyczne parametry jak funkcji glTexImage1D. Dodatkowo wymaga podania jedynie argumentu height oraz jako target GL_TEXTURE_2D.

```
void glTexImage2D(GLenum target, GLint level, GLint components,  
                 GLsizei width, GLsizei height, GLint border, GLenum format, GLenum type, const GLvoid  
                 *pixels)
```

Parametr `pixels` wskazuje na dane pikseli w postaci binarnej i jest niezależny od formatu tekstury na dysku (można nawet ręcznie stworzyć ją w kodzie programu). Z tego względu należy napisać własną funkcję do wczytywania i konwersji (przykład został załączony w oddzielnym pliku).

1.3 Nakładanie tekstur

Nakładanie tekstur wykonuje się za pomocą rodziny funkcji `glTexCoord`. W zależności od rodzaju tekstury funkcja ta może przyjmować jeden lub dwa parametry określające punkt tekstury odpowiadający kolejnemu definiowanemu przez polecenie `glVertex` wierzchołkowi.

Przykład:

```
glBegin(GL_TRIANGLES);

    glTexCoord2f(0.1f, 0.0f);

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

    glTexCoord2f(1.0f, 0.0f);

    glVertex3f(2.0f,0.0f,0.0f);

    glTexCoord2f(0.1f, 1.0f);

    glVertex3f(0.0f,-2.0f,0.0f);

glEnd();
```

1.4 Filtry tekstur

OpenGL pozwala na zwiększanie lub zmniejszanie tekstur. Filtr pomniejszający wykorzystywany jest w przypadku, gdy obraz przypisany do tekstury na ekranie jest mniejszy niż w rzeczywistości. Natomiast filtr powiększający wykorzystywany jest gdy obraz z tekstury na ekranie jest większy niż wczytany z pliku.

Funkcją, którą możemy ustawiać parametry tekstur oraz rodzaj filtrowania jest `glTexParameter`.

```
void glTexParameterf(GLenum target, GLenum pname, GLfloat param)

void glTexParameteri(GLenum target, GLenum pname, GLint param)
```

Przyjmuje ona trzy parametry. Pierwszy parametr określa typ tekstury (`GL_TEXTURE_1D`, `GL_TEXTURE_2D`). Drugi to nazwa zmienianego parametru. Ostatni parametr to nowa wartość zmienianego parametru tekstury.

Parametr `pname` może posiadać jedną z wartości: `GL_TEXTURE_MIN_FILTER`,

GL_TEXTURE_MAG_FILTER, GL_TEXTURE_WRAP_S, GL_TEXTURE_WRAP_T,
GL_TEXTURE_BORDER_COLOR, GL_TEXTURE_PRIORITY.

Aby ustawić filtrowanie pomniejszające jako drugi parametr funkcji glTexParameter musimy ustawić GL_TEXTURE_MIN_FILTER, a dla filtrowania powiększającego GL_TEXTURE_MAG_FILTER. Filtrowanie w obu przypadkach może przyjmować następujące wartości, które przekazujemy jako trzeci parametr funkcji glTexParameter.

Parametr „param”	Opis
GL_NEAREST	Używa wartości najbliższego teksla renderowanego piksela
GL_LINEAR	Stosuje interpolację liniową między czterema tekstami
GL_NEAREST_MIPMAP_NEAREST	Używa filtru GL_NEAREST oraz mipmapy o rozdzielczości najbardziej zbliżonej do rozdzielczości piksela
GL_NEAREST_MIPMAP_LINEAR	Używa filtru GL_NEAREST oraz stosuje interpolację liniową
GL_LINEAR_MIPMAP_NEAREST	Używa filtru GL_LINEAR oraz mipmapy o rozdzielczości najbardziej zbliżonej do rozdzielczości piksela
GL_LINEAR_MIPMAP_LINEAR	Używa filtru GL_LINEAR oraz stosuje interpolację liniową

2 Zadanie

Napisz program, w którym za pomocą sześciątów narysujesz skrzynki lub pudełka kartonowe w trzech wymiarach. Zastosuj minimum dwie różne tekstury (np. stworzone w programie Gimp), do wczytywania tekstur z pliku możesz użyć dołączonego pliku.