FONDAMENTI DI COMPUTER GRAPHICS M

LAB 07

Il settimo laboratorio ha riguardato gestione di texture mapping, normal mapping ed environment mapping con OpenGL e GLSL. È stata fornita una scena con alcuni oggetti (toro, colonna, pietra, sfera, cubo, finestre, …) da cui partire. Il laboratorio può essere diviso in tre parti principali:

1. Applicazioni di semplici texture grafiche, sia a partire da immagini, sia generate proceduralmente;
2. Applicazione del multitexturing con texture mapping e normal mapping;
3. Environment mapping, ovvero la riproduzione di texture ambientali sugli oggetti riflettenti o “trasparenti” (refraction).

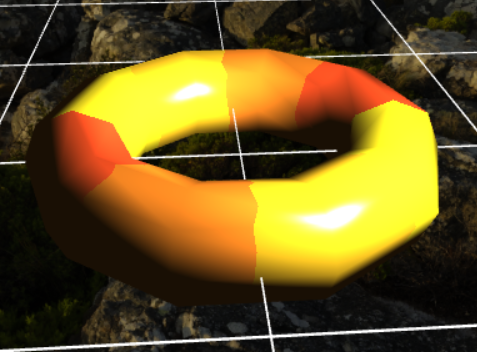
**1. Texture mapping, shading e texture procedurali**

Il primo step è stato quello di capire come usare OpenGL (e GLSL) per applicare texture, ovvero immagini, agli oggetti presenti in scena. Per farlo, è necessario prima di tutto caricare l’immagine voluta in un array che contiene le informazioni utili, come i canali RGB. Dopo aver applicato alcuni parametri alla texture (come per esempio filtri, tramite la funzione glTexParameteri), sarà necessario caricare la texture nella memoria della GPU tramite glTexImage2D. A questo punto, basterà associare la texture caricata all’oggetto (tramite id – la texture viene caricata una sola volta, ma può essere utilizzata più volte).

Nel caso sopra descritto, il vertex shader si limita a prendere in input le varie informazioni (matrici) per le trasformazioni e le coordinate texture, passando queste ultime al fragment shader, che, avendo deciso per un texture mapping molto semplice, si limiterà a sostituire il colore del fragment con quello della coordinata texture.

A questo punto, è possibile aggiungere l’illuminazione così come visto nei laboratori precedenti attraverso gli shader (specificando per ogni oggetto, quindi, anche il tipo di materiale, che imposterà il comportamento dell’oggetto rispetto alla luce). Si applica Phong Shading: nel vertex shader si calcolaranno i tre vettori E, N e L, oltre a passare le coordinate texture, mentre nel fragment la texture sarà sostituita alla componente diffusiva (kd) nella formula per il shading di tipo phong.

Infine, è stato scritto un piccolo algoritmo per la generazione di una texture procedurale (a strisce colorate di 4 colori diversi). Tale algoritmo riempie una matrice di dimensione prefissata e di 3 canali RGB, che andrà a sostituire quella caricata da immagine.



2. Multi-texturing

In questa parte di laboratorio, l’obiettivo è stato quello di

