**PRAKTIKUM 4.**

**PENGENALAN TOOLS : PEMETAAN TEKSTUR DAN TEKSTUR  TRANSPARAN**

**1.1 TUJUAN PRAKTIKUM**

**Tujuan Umum**

1. Memahami konsep pemetaan tekstur dalam grafika komputer.

2. Mengerti bagaimana menerapkan tekstur pada objek 3D.

3. Mengetahui penggunaan tekstur transparan dalam grafika komputer.

4. Mampu mengimplementasikan pemetaan tekstur dan tekstur transparan pada suatu  objek.

**Tujuan Khusus**

1. Mempelajari prinsip dasar pemetaan tekstur.

2. Memahami langkah-langkah untuk memetakan tekstur pada permukaan objek 3D  menggunakan OpenGL.

3. Mengetahui konsep tekstur transparan.

4. Menguasai teknik pemetaan warna dan tekstur pada berbagai sisi objek 3D. 5. Dapat mengaplikasikan tekstur transparan.

**1.2 PENGENALAN PEMETAAN TEKSTUR DAN TEKSTUR TRANSPARAN Pemetaan Tekstur (Texture Mapping):**

Merupakan pemberian detail visual pada objek 3D dengan menggunakan gambar atau  warna pada permukaan objek. Mirip dengan konsep menempelkan stiker/warna pada  objek, pemetaan tekstur menciptakan tampilan yang lebih.

**Tekstur Transparan (Transparent Textures):**

Jenis tekstur ini memungkinkan sebagian dari objek menjadi transparan atau tembus  cahaya.Penggunaan saluran (kanal transparansi) pada tekstur transparan menentukan  tingkat transparansi, dan implementasi yang dapat meningkatkan kompleksitas visual  dalam suatu grafis.

**1.3 PELAKSANAAN PRAKTIKUM**

**1.3.1.PEMETAAN TEKSTUR**

**initRendering():**

untuk membuat dan memetakan tekstur warna untuk masing-masing sisi objek.  **glEnable(GL\_DEPTH\_TEST);** (untuk mengaktifkan kedalaman warna)  **glEnable(GL\_LIGHTING);** (untuk pencahayaan)

**glEnable(GL\_LIGHT0);** (untuk pencahayaan dengan parameter)

**glEnable(GL\_NORMALIZE);** (untuk semua teksture tidak abnormal)

**glEnable(GL\_COLOR\_MATERIAL);** (untuk warna dari setiap material) **loadColorTexture(const GLfloat color[], GLuint &textureId):**

Fungsi untuk membuat tekstur warna berdasarkan array warna dan menghasilkan  ID tekstur.

**glGenTextures** (untuk menghasilkan ID tekstur baru)

**glBindTexture** (untuk floating tekstur dengan ID yang diberikan)

**glTexParameteri** (untuk parameter tekstur LINEAR)

**glTexImage2D** (untuk parameter tekstur dan sebagai citra tekstur ) **drawScene():**

Fungsi untuk menggambar sebuah objek

**glBindTexture(GL\_TEXTURE\_2D, \_textureId)** (untuk mengikat tekstur dan  objek)

**glEnable(GL\_TEXTURE\_2D)** (definisi untuk tekstur 2D)

**glEnable(GL\_LIGHTING)** (memastikan pemetaan cahaya aktfi)

Gambar masing-masing sisi objek dengan pemetaan tekstur

• **glBegin(GL\_QUADS)**

• **glEnd()**

**handleResize(int w, int h):**

berkaitan dengan perubahan ukuran layar.

**glViewport**

▪ Fungsi ini mengatur area tampilan di dalam jendela OpenGL.

▪ Parameter x dan y menentukan posisi sudut kiri bawah viewport,  sedangkan width dan height menentukan lebar dan tinggi viewport.

**glMatrixMode**

• Fungsi ini digunakan untuk mengatur matriks yang sedang dioperasikan,  seperti matriks modelview, matriks proyeksi, atau matriks tekstur.

• Parameter **mode** menentukan matriks mana yang akan dioperasikan,  seperti GL\_MODELVIEW, GL\_PROJECTION, atau GL\_TEXTURE.

**glLoadIdentity**

• Fungsi ini mengganti matriks yang aktif menjadi matriks identitas.

• Matriks identitas adalah matriks yang tidak menghasilkan transformasi apa  pun.

**gluPerspective**

Fungsi ini mengatur matriks proyeksi untuk proyeksi perspektif.

▪ fovy adalah sudut pandang vertical dalam derajat.

▪ aspect adalah rasio aspek (lebar jendela dibagi tinggi jendela).

▪ adalah jarak ke depan dan ke belakang dari kamera

**1.3.2.TEKSTUR TRANSPARAN**

**GLfloat ambientLight[]**

• Menginisialisasi array ambientLight dengan nilai-nilai float

• Format warna RGBA (Red, Green, Blue, dan Alpha), di mana nilai alpha adalah  tingkat transparansi. Dalam hal ini nilai alpha adalah 1.0f, yang berarti tidak ada  transparansi.

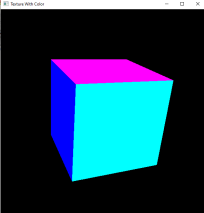
**glLightModelfv(GL\_LIGHT\_MODEL\_AMBIENT, ambientLight)**

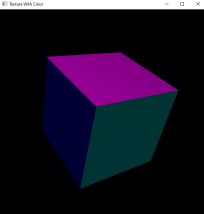
• Mengatur model cahaya ambient untuk seluruh objek.

• Parameter pertama **GL\_LIGHT\_MODEL\_AMBIENT** menentukan untuk  mengatur properti cahaya ambient secara keseluruhan.

• Parameter kedua **ambientLight** adalah array yang berisi nilai-nilai warna.

**1.3.3.HASIL RUNNING PROGRAM PEMETAAN TEKSTUR**

**TEKSTUR TRANSPARAN**

****