



FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

LAB. SHEET PRAKTIKUM GRAFIKA KOMPUTER 1

No. : ST/EKA/PTI223/06

Revisi : 02

Senin 010509

Hal. 1 dari 9 hal.

Sem.: Genap

Pengkabutan pada Grafik 3D

4 x 50 menit

Pengantar

Fog adalah pengaruh atmosferic yang ditambahkan objek, sehingga akan terjadi kekaburan pada obyek yang tergantung seberapa jauh benda tersebut dari pandangan orang yang melihatnya. Biasanya fog biasanya dibahas bersamaan dengan Blending dan Antialiasing. Dengan membahas ketiga topik tersebut, dapat digunakan untuk menambahkan sedikit realisme benda (benda yang kelihatan nyata). Menggunakan blending dapat mendukung efek seperti transparansi yang dapat mensimulasikan gelas minuman, jendela dan transparan benda lainnya. Kabut dapat berfungsi untuk mensimulasikan kondisi cuaca seperti hujan, kabut dan lain-lain dengan mengurangi visibilitas dan jarak. Fog dapat digunakan untuk membuat pemandangan dengan efek atmosferic realistis. Menggunakan antialiasing, dapat digunakan untuk menghaluskan obyek yang bergerigi atau kasar pada bagian tertentu.

Sebuah gambar dapat dibuat agar tampak lebih alami dengan menambahkan sebuah kabut, yang membuatnya menjadi objek terpengaruh jarak jarak. Kabut secara umum adalah istilah yang menggambarkan bentuk berupa atmosferic efek, yang dapat digunakan untuk mensimulasikan asap rokok, dan atau polusi. Kabut sangat penting dalam simulasi visual-aplikasi, tempat terbatas pada pendekatan visibilitas.

Bila kabut diaktifkan, objek yang jauh dari pandangan mulai berangsur-angsur masuk ke dalam warna/tertutup kabut. Anda dapat mengontrol kepadatan dari kabut, yang menentukan harga pada obyek yang secara perlahan sebagai jarak meningkat, serta kabut warna. setelah kabut diterapkan dengan kombinasi transformasi matrik, pencahayaan, dan texturing dilakukan, maka akan mempengaruhi obyek, lit, dan texture sebuah obyek benda.

Menggunakan kabut sangat mudah. Mengaktifkannya lewat oleh `GL_FOG` ke `glEnable()`, dan kita bisa juga memilih warna dan persamaan untuk mengendalikan kepadatan dengan `glFog*()`. Kabut dapat ditetapkan dengan mengaktifkan kabut dan mengatur fungsi kabut seperti yang ditunjukkan di bawah ini:

```
glFogi(GL_FOG_MODE, GL_EXP);
```

`GL_EXP` adalah mode kabut untuk kabut pada mendung/awan.

```
GLfloat fog_color[4] = {0.2,0.2,0.2,0.0};
```

Pengaturan warna dan kepadatan

```
glFogfv(GL_FOG_COLOR, fog_color);  
glFogf(GL_FOG_DENSITY, 0.25);
```

Perhatikan contoh program dibawah ini:



FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

LAB. SHEET PRAKTIKUM GRAFIKA KOMPUTER 1

No. : ST/EKA/PTI223/06

Revisi : 02

Senin 010509

Hal. 2 dari 9 hal.

Sem.: Genap

Pengkabutan pada Grafik 3D

4 x 50 menit

Program 1

```
#include <GL/glut.h>
```

```
#include <math.h>
```

```
#include <stdlib.h>
```

```
#include <stdio.h>
```

```
static GLint fogMode;
```

```
/* inisialisasi kedalaman buffer, fog, sumber cahaya, sifat-sifat bahan dan model pencahayaan */
```

```
static void init(void)
```

```
{
```

```
    GLfloat position[] = { 0.5, 0.5, 3.0, 0.0 };
```

```
    glEnable(GL_DEPTH_TEST);
```

```
    glLightfv(GL_LIGHT0, GL_POSITION, position);
```

```
    glEnable(GL_LIGHTING);
```

```
    glEnable(GL_LIGHT0);
```

```
{
```

```
    GLfloat mat[3] = {0.1745, 0.01175, 0.01175};
```

```
    glMaterialfv (GL_FRONT, GL_AMBIENT, mat);
```

```
    mat[0] = 0.61424; mat[1] = 0.04136; mat[2] = 0.04136;
```

```
    glMaterialfv (GL_FRONT, GL_DIFFUSE, mat);
```

```
    mat[0] = 0.727811; mat[1] = 0.626959; mat[2] = 0.626959;
```

```
    glMaterialfv (GL_FRONT, GL_SPECULAR, mat);
```

```
    glMaterialf (GL_FRONT, GL_SHININESS, 0.6*128.0);
```

```
}
```

```
    glEnable(GL_FOG);
```

```
{
```

```
    GLfloat fogColor[4] = {0.5, 0.5, 0.5, 1.0};
```

```
    fogMode = GL_EXP;
```

```
    glFogi (GL_FOG_MODE, fogMode);
```

```
    glFogfv (GL_FOG_COLOR, fogColor);
```

```
    glFogf (GL_FOG_DENSITY, 0.35);
```

```
    glHint (GL_FOG_HINT, GL_DONT_CARE);
```

```
    glFogf (GL_FOG_START, 1.0);
```

```
    glFogf (GL_FOG_END, 5.0);
```

```
}
```

```
    glClearColor(0.5, 0.5, 0.5, 1.0); /* memberi warna kabut */
```

```
}
```



FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

LAB. SHEET PRAKTIKUM GRAFIKA KOMPUTER 1

No. : ST/EKA/PTI223/06

Revisi : 02

Senin 010509

Hal. 3 dari 9 hal.

Sem.: Genap

Pengkabutan pada Grafik 3D

4 x 50 menit

static void renderSphere (GLfloat x, GLfloat y, GLfloat z)

```
{
    glPushMatrix();
    glTranslatef (x, y, z);
    glutSolidSphere(0.4, 16, 16);
    glPopMatrix();
}
```

/ display() untuk menampilkan gambar 5 sphere pada posisi z yang berbeda. */*

void display(**void**)

```
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    renderSphere (-2., -0.5, -1.0);
    renderSphere (-1., -0.5, -2.0);
    renderSphere (0., -0.5, -3.0);
    renderSphere (1., -0.5, -4.0);
    renderSphere (2., -0.5, -5.0);
    glFlush();
}
```

void reshape(**int** w, **int** h)

```
{
    glViewport(0, 0, (GLsizei) w, (GLsizei) h);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    if (w <= h)
        glOrtho (-2.5, 2.5, -2.5*(GLfloat)h/(GLfloat)w,
                2.5*(GLfloat)h/(GLfloat)w, -10.0, 10.0);
    else
        glOrtho (-2.5*(GLfloat)w/(GLfloat)h,
                2.5*(GLfloat)w/(GLfloat)h, -2.5, 2.5, -10.0, 10.0);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity ();
}
```

void keyboard(**unsigned char** key, **int** x, **int** y)

```
{
    switch (key) {
        case 'f':
        case 'F':
            if (fogMode == GL_EXP) {
                fogMode = GL_EXP2;
                printf ("Kabut dalam mode GL_EXP2\n");
            }
    }
}
```



FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

LAB. SHEET PRAKTIKUM GRAFIKA KOMPUTER 1

No. : ST/EKA/PTI223/06

Revisi : 02

Senin 010509

Hal. 4 dari 9 hal.

Sem.: Genap

Pengkabutan pada Grafik 3D

4 x 50 menit

```
}
else if (fogMode == GL_EXP2) {
    fogMode = GL_LINEAR;
    printf ("Kabut dalam mode GL_LINEAR\n");
}
else if (fogMode == GL_LINEAR) {
    fogMode = GL_EXP;
    printf ("Kabut dalam mode GL_EXP\n");
}
glFogi (GL_FOG_MODE, fogMode);
glutPostRedisplay();
break;
case 27:
    exit(0);
    break;
default:
    break;
}
}
}

int main(int argc, char** argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB | GLUT_DEPTH);
    glutInitWindowSize(500, 500);
    glutCreateWindow(argv[0]);
    init();
    glutReshapeFunc (reshape);
    glutKeyboardFunc (keyboard);
    glutDisplayFunc (display);
    glutMainLoop();
    return 0;
}
```

Program 2

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <GL/glut.h>
```

```
GLenum doubleBuffer;
```

```
double plane[4] = {1.0, 0.0, -1.0, 0.0};
float rotX = 5.0, rotY = -5.0, zTranslate = -65.0;
```



FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

LAB. SHEET PRAKTIKUM GRAFIKA KOMPUTER 1

No. : ST/EKA/PTI223/06

Revisi : 02

Senin 010509

Hal. 5 dari 9 hal.

Sem.: Genap

Pengkabutan pada Grafik 3D

4 x 50 menit

```
float fogDensity = 0.02;
GLint cubeList = 1;
```

```
float scp[18][3] = {
    {1.000000, 0.000000, 0.000000},
    {1.000000, 0.000000, 5.000000},
    {0.707107, 0.707107, 0.000000},
    {0.707107, 0.707107, 5.000000},
    {0.000000, 1.000000, 0.000000},
    {0.000000, 1.000000, 5.000000},
    {-0.707107, 0.707107, 0.000000},
    {-0.707107, 0.707107, 5.000000},
    {-1.000000, 0.000000, 0.000000},
    {-1.000000, 0.000000, 5.000000},
    {-0.707107, -0.707107, 0.000000},
    {-0.707107, -0.707107, 5.000000},
    {0.000000, -1.000000, 0.000000},
    {0.000000, -1.000000, 5.000000},
    {0.707107, -0.707107, 0.000000},
    {0.707107, -0.707107, 5.000000},
    {1.000000, 0.000000, 0.000000},
    {1.000000, 0.000000, 5.000000},
};
```

```
static float ambient[] = {0.1, 0.1, 0.1, 1.0};
static float diffuse[] = {1.0, 1.0, 1.0, 1.0};
static float position[] = {90.0, 90.0, 0.0, 0.0};
static float front_mat_shininess[] = {30.0};
static float front_mat_specular[] = {0.0, 0.0, 0.0, 1.0};
static float front_mat_diffuse[] = {0.0, 1.0, 0.0, 1.0};
static float back_mat_shininess[] = {50.0};
static float back_mat_specular[] = {0.0, 0.0, 1.0, 1.0};
static float back_mat_diffuse[] = {1.0, 0.0, 0.0, 1.0};
static float lmodel_ambient[] = {0.0, 0.0, 0.0, 1.0};
static float fog_color[] = {0.8, 0.8, 0.8, 1.0};
```

```
static void Key(unsigned char key, int x, int y)
{
    switch (key) {
        case 'd':
            fogDensity *= 1.10;
            glFogf(GL_FOG_DENSITY, fogDensity);
    }
}
```



FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

LAB. SHEET PRAKTIKUM GRAFIKA KOMPUTER 1

No. : ST/EKA/PTI223/06

Revisi : 02

Senin 010509

Hal. 6 dari 9 hal.

Sem.: Genap

Pengkabutan pada Grafik 3D

4 x 50 menit

```
glutPostRedisplay();
break;
case 'D':
    fogDensity /= 1.10;
    glFogf(GL_FOG_DENSITY, fogDensity);
    glutPostRedisplay();
    break;
case 27:
    exit(0);
}
}
```

```
static void SpecialKey(int key, int x, int y)
{
    switch (key) {
    case GLUT_KEY_UP:
        rotX -= 5;
        glutPostRedisplay();
        break;
    case GLUT_KEY_DOWN:
        rotX += 5;
        glutPostRedisplay();
        break;
    case GLUT_KEY_LEFT:
        rotY -= 5;
        glutPostRedisplay();
        break;
    case GLUT_KEY_RIGHT:
        rotY += 5;
        glutPostRedisplay();
        break;
    }
}
```

```
static void
Draw(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    glPushMatrix();

    glTranslatef(0, 0, zTranslate);
    glRotatef(rotY, 0, 1, 0);
    glRotatef(rotX, 1, 0, 0);
```



FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

LAB. SHEET PRAKTIKUM GRAFIKA KOMPUTER 1

No. : ST/EKA/PTI223/06

Revisi : 02

Senin 010509

Hal. 7 dari 9 hal.

Sem.: Genap

Pengkabutan pada Grafik 3D

4 x 50 menit

```
glScalef(1.0, 1.0, 10.0);
```

```
glCallList(cubeList);
```

```
glPopMatrix();
```

```
if (doubleBuffer) {  
    glutSwapBuffers();  
} else {  
    glFlush();  
}  
}
```

```
static void
```

```
Args(int argc, char **argv)
```

```
{  
    GLint i;  
    doubleBuffer = GL_TRUE;  
  
    for (i = 1; i < argc; i++) {  
        if (strcmp(argv[i], "-sb") == 0) {  
            doubleBuffer = GL_FALSE;  
        } else if (strcmp(argv[i], "-db") == 0) {  
            doubleBuffer = GL_TRUE;  
        }  
    }  
}
```

```
int main(int argc, char **argv)
```

```
{  
    GLenum type;  
  
    glutInit(&argc, argv);  
    Args(argc, argv);  
  
    type = GLUT_RGB | GLUT_DEPTH;  
    type |= (doubleBuffer) ? GLUT_DOUBLE : GLUT_SINGLE;  
    glutInitDisplayMode(type);  
    glutInitWindowSize(300, 300);  
    glutCreateWindow("Test Kabut");  
  
    glFrontFace(GL_CW);  
  
    glEnable(GL_DEPTH_TEST);
```



FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

LAB. SHEET PRAKTIKUM GRAFIKA KOMPUTER 1

No. : ST/EKA/PTI223/06

Revisi : 02

Senin 010509

Hal. 8 dari 9 hal.

Sem.: Genap

Pengkabutan pada Grafik 3D

4 x 50 menit

```
glLightfv(GL_LIGHT0, GL_AMBIENT, ambient);
glLightfv(GL_LIGHT0, GL_DIFFUSE, diffuse);
glLightfv(GL_LIGHT0, GL_POSITION, position);
glLightModelfv(GL_LIGHT_MODEL_AMBIENT, lmodel_ambient);
glLightModeli(GL_LIGHT_MODEL_TWO_SIDE, GL_TRUE);
glEnable(GL_LIGHTING);
glEnable(GL_LIGHT0);
```

```
glMaterialfv(GL_FRONT, GL_SHININESS, front_mat_shininess);
glMaterialfv(GL_FRONT, GL_SPECULAR, front_mat_specular);
glMaterialfv(GL_FRONT, GL_DIFFUSE, front_mat_diffuse);
glMaterialfv(GL_BACK, GL_SHININESS, back_mat_shininess);
glMaterialfv(GL_BACK, GL_SPECULAR, back_mat_specular);
glMaterialfv(GL_BACK, GL_DIFFUSE, back_mat_diffuse);
```

```
glEnable(GL_FOG);
glFogf(GL_FOG_MODE, GL_EXP);
glFogf(GL_FOG_DENSITY, fogDensity);
glFogfv(GL_FOG_COLOR, fog_color);
glClearColor(0.8, 0.8, 0.8, 1.0);
```

```
glNewList(cubeList, GL_COMPILE);
glBegin(GL_TRIANGLE_STRIP);
    glNormal3fv(scp[0]);
    glVertex3fv(scp[0]);
    glNormal3fv(scp[0]);
    glVertex3fv(scp[1]);
    glNormal3fv(scp[2]);
    glVertex3fv(scp[2]);
    glNormal3fv(scp[2]);
    glVertex3fv(scp[3]);
    glNormal3fv(scp[4]);
    glVertex3fv(scp[4]);
    glNormal3fv(scp[4]);
    glVertex3fv(scp[5]);
    glNormal3fv(scp[6]);
    glVertex3fv(scp[6]);
    glNormal3fv(scp[6]);
    glVertex3fv(scp[7]);
    glNormal3fv(scp[8]);
    glVertex3fv(scp[8]);
    glNormal3fv(scp[8]);
    glVertex3fv(scp[9]);
```




FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

LAB. SHEET PRAKTIKUM GRAFIKA KOMPUTER 1

No. : ST/EKA/PTI223/06

Revisi : 02

Senin 010509

Hal. 9 dari 9 hal.

Sem.: Genap

Pengkabutan pada Grafik 3D

4 x 50 menit

```
glNormal3fv(scp[10]);
glVertex3fv(scp[10]);
glNormal3fv(scp[10]);
glVertex3fv(scp[11]);
glNormal3fv(scp[12]);
glVertex3fv(scp[12]);
glNormal3fv(scp[12]);
glVertex3fv(scp[13]);
glNormal3fv(scp[14]);
glVertex3fv(scp[14]);
glNormal3fv(scp[14]);
glVertex3fv(scp[15]);
glNormal3fv(scp[16]);
glVertex3fv(scp[16]);
glNormal3fv(scp[16]);
glVertex3fv(scp[17]);
glEnd();
glEndList();
```

```
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
gluPerspective(45.0, 1.0, 1.0, 200.0);
glMatrixMode(GL_MODELVIEW);
```

```
glutKeyboardFunc(Key);
glutSpecialFunc(SpecialKey);
glutDisplayFunc(Draw);
glutMainLoop();
return 0;
}
```

Tugas

1. Cobalah program diatas
2. Lakukan perubahan sintak atau nilai yang ada dalam program diatas (yang berhubungan dengan kabut saja)
3. Masukkan hasil pengamatan dan masukan hasilnya yang disertai gambar hasil compile dalam tabel pengamatan yang disesuaikan dengan pengamatan anda.
4. Buatlah gambar segiempat 3D (kubus) yang diselimuti kabut
5. Buat gambar sebuah gambar obyek (bebas) yang dilengkapi dengan pencahayaan dan pengkabutan.