BAB II

LANDASAN TEORI

2.1 Grafika Komputer

Grafika komputer adalah seperangkat alat yang terdiri dari *hardware* dan *software* untuk membuat gambar, grafik, atau citra realistik untuk seni, game komputer, foto dan film animasi.[1]

Hill (1990:2) menyatakan bahwa grafika komputer adalah sekumpulan alat yang digunakan untuk membuat gambar (*to create picture)* dan berinteraksi dengan gambar dengan cara-cara seperti biasa digunakan (*and to interact with them in natural ways)*. Peralatan tersebut dapat berupa perangkat keras (*hardware*) maupun perangkat lunak (*software*). Semua peralatan tersebut memberi fasilitas untuk membuat program grafis yang baik. [2]

Secara umum program grafis ditujukan untuk memudahkan interaksi antara manusia dan komputer. Dengan memvisualkan kata-kata menjadi gambar maka manusia akan menjadi lebih mudah berinteraksi dengan komputer, perintah-perintah tak lagi diberikan dengan cara menulis tetapi dengan cara menunjuk area atau gambar tertentu. [3]

Grafika Komputer secara umum dapat dibagi menjadi dua bagian, yaitu grafika 2 dimensi dan grafika 3 dimensi. Dalam teknik penyajian, grafika 2 dimensi memiliki kekurangan, yaitu ketidakmampuan untuk merepresentasikan objek 3 dimensi. Pemodelan 3 dimensi tidak dapat memenuhi kebutuhan itu, sehingga mendorong dikembangkannya grafika 2 dimensi.

Grafika 3 dimensi dapat menggambarkan sebuah benda dalam dunia nyata menjadi sebuah objek dari berbagai sudut pandang. Tidak seperti grafika 2 dimensi, dalam grafika 3 dimensi banyak sekali hal yang harus dipertimbangkan.

2.2 Dasar Animasi

Animasi pada dasarnya adalah membuat objek seolah-olah bergerak sehingga terjadi pergerakan yang membuat mata kita memandangnya sebagai sebuah kesatuan utuh .padahal dibalik itu, animasi dibuat berdasarkan bagian-bagian tertentu (terpisah) dan bari kemudian akan disatukan untuk mendapatkan pergerakan dengan sempurna.

Untuk membuat animasi, terlebih dahulu kita harus membuat satu persatu bagian-bagian tertentu tadi. Hal ini merupakan hal yang sangat berat dalam membuat animasi mengingat kita harus memikirkan bagaimana desain atau bagian dari animasi yang kita buat tersebut supaya benar-benar mendapatkan hasil yang sempurna jika disatukan.

2.2.1 Elemen-Elemen Grafika Komputer

2.2.1.1 Objek 3 Dimensi

Secara konseptual dalam grafika komputer objek 3 dimensi dibuat dari rangkaian poligon. Poligon adalah sejumlah dari segmen garis yang dihubungkan antara ujung garis satu dengan yang lainnya membentuk suatu bidang tertutup.

Objek 3 dimensi memiliki tiga ukuran yaitu panjang, lebar dan kedalaman, dimana dalam geometri disimbolkan X, Y dan Z. Menurut fungsinya objek 3

dimensi mempunyai dua variable bebas (X dan Y) dan satu variable tidak bebas (Z, karena nilai Z tergantung pada nilai X dan Y). Objek 3 dimensi diantaranya sebagai berikut :

1. Kubus

   Kubus adalah balok dengan bidang-bidang sisi yang bujur sangkar. Batas-batas luar kubus membentuk poligon enam sisi (heksagonal).

2. Balok

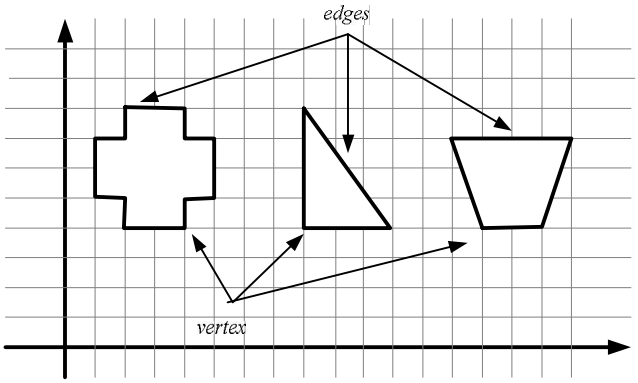
   Balok adalah poligon tiga sisi (heksagonal) beraturan dengan bidang sisi terdiri dari dua buah persegi panjang dan dua buah bujur sangkar.

3. Bola

   Bentuk bola diperoleh dengan memutar busur ½ lingkaran melalui sumbu yang melalui titik pusatnya.

2.2.1.2 Poligon

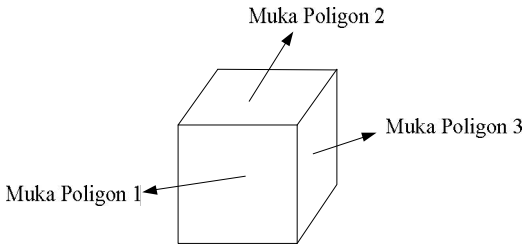
Poligon adalah bentuk yang disusun dari serangkaian garis. Gambar 2.3 memberikan beberapa contoh poligon. Titik sudut dari poligon disebut *vertex* sedangkan garis penyusun poligon disebut *edge*. [4]



Gambar 2.3 Poligon

Poligon digambar dengan menggambar masing-masing *edge* dengan setiap *edge* merupakan pasangan dari $vertex_i$ – $vertex_{i+1}$ kecuali untuk *edge* terakhir merupakan pasangan dari $vertex_n$ – $vertex_1$. [3]

Jaring Poligon adalah permukaan yang dibuat dari kumpulan muka-muka poligon. Secara konseptual, jaring poligon digunakan di grafika komputer untuk menghasilkan objek 3 dimensi. Untuk lebih jelasnya, lihat pada gambar 2.4.



Gambar 2.4 Ilustrasi muka poligon dengan objek kubus

2.3 Transformasi Objek 3 Dimensi

Salah satu sub bagian dari grafika komputer adalah pemodelan objek (*object modeling*). Dalam pemodelan objek 2 dimensi, didapati berbagai objek

dapat dimodelkan. Menurut kondisi tertentu, objek yang dimodelkan itu perlu dimodifikasi. Pemodifikasian objek ini dapat dilakukan dengan melakukan berbagai operasi fungsi atau operasi transformasi geometri. Transformasi ini dapat berupa transformasi dasar ataupun gabungan dari berbagai transformasi geometri. Pada dasarnya transformasi adalah memindahkan objek tanpa merusak bentuk.

Tujuan transformasi adalah :

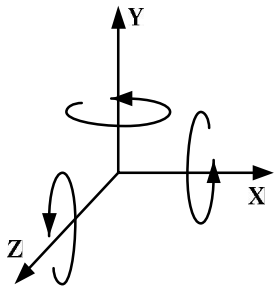
1. Merubah atau menyesuaikan komposisi pemandangan.
2. Memudahkan membuat objek yang simetris.
3. Melihat objek dari sudut pandang yang berbeda.
4. Memindahkan satu atau beberapa objek dari satu tempat ke tempat lain, ini biasa dipakai untuk animasi komputer.

Transformasi pada dunia 3 dimensi, pada dasarnya sama dengan transformasi pada 2 dimensi, hanya pada 3 dimensi kita juga memperhitungkan sumbu Z. Sama seperti pada 2 dimensi, ada tiga transformasi dasar yang dapat dilakukan yaitu translasi, penskalaan, rotasi. Perbedaannya adalah pada objek 3 dimensi proses transformasinya dilakukan dengan mempertimbangkan koordinat yang merupakan besarnya kedalaman dari objek.

2.3.1 Rotasi

Berbeda dengan rotasi 2 dimensi yang menggunakan titik pusat (0,0) sebagai pusat perputaran, rotasi 3 dimensi menggunakan sumbu koordinat sebagai pusat perputaran.Dengan demikian ada tiga macam rotasi yang dapat dilakukan,

yaitu : rotasi sumbu X, rotasi sumbu Y, rotasi sumbu Z. Untuk lebih jelasnya, lihat pada gambar 2.4. [4]



Gambar 2.4 Rotasi dan Sumbu Rotasi

Sebuah titik pada system koordinat dapat diputar pada setiap sumbu yang disebut rotasi. Sudut rotasi adalah besaran yang menentukan seberapa jauh sebuah titik akan diputar. Sudut rotasi pada sumbu X, Y, dan Z masing-masing akan dinyatakan dalam besaran r_x, r_y, dan r_z.

Pada sumbu X dengan sudut r_x, adalah :

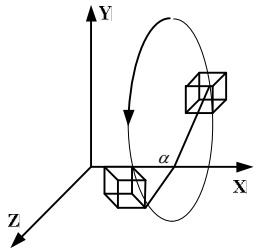
$$Y' = Y\cos\alpha - Z\sin\alpha$$
$$Z' = Y\sin\alpha + Z\cos\alpha$$
$$X' = X$$

(2-1)

Rotasi pada sumbu X dinyatakan dalam bentuk matriks homogen sebagai berikut :

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Dengan gambar seperti pada gambar 2.5 berikut :



Gambar 2.5 Rotasi pada sumbu X

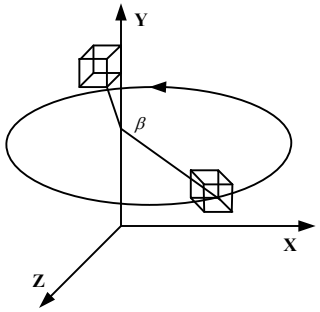
Pada sumbu Y adalah :

$$\begin{aligned} Z' &= Z\cos\beta - X\sin\beta \\ X' &= Z\sin\beta + X\cos\beta \\ Y' &= Y \end{aligned} \qquad (2\text{-}2)$$

Rotasi pada sumbu Y dinyatakan dalam bentuk matriks homogen sebagai berikut :

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Dengan gambar seperti pada gambar 2.6 berikut :



Gambar 2.6 Rotasi pada sumbu Y

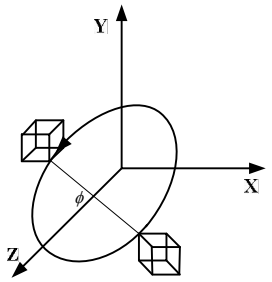
Rotasi pada sumbu Z dengan sudut r_z, menggunakan persamaan :

$$X' = X\cos\phi - Y\sin\phi$$
$$Y' = X\sin\phi + Y\cos\phi \qquad (2\text{-}3)$$
$$Z' = Z$$

Rotasi pada sumbu Z dinyatakan dalam matriks homogen sebagai berikut :

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Dengan gambar seperti pada gambar 2.7 berikut :



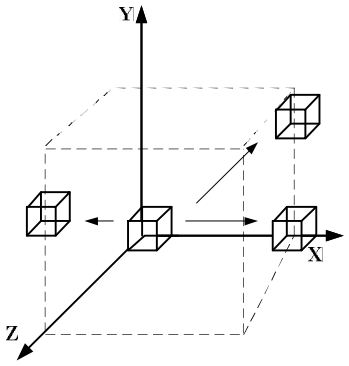
Gambar 2.7 Rotasi pada sumbu Z

Persamaan-persamaan di atas adalah persamaan untuk mendapatkan titik hasil rotasi dari titik awalnya.

2.1.3.2 Translasi

Suatu objek yang tersapat didalam sistem koordinat ruang 3 dimensi dapat dipindahkan dari posisi awalnya sesuai dengan yang diinginkan selama objek tersebut masih terdapat dalam sistem koordinat ruang 3 dimensi. Secara teoritis jarak geser tidak terbatas, tetapi pada kenyataannya untuk memudahkan

identifikasi keberadaan suatu objek ditentukan batas maksimumnya untuk mengkonversi objek dari sistem koordinat ke sistem koordinat layar, karena sistem koordinat layar memiliki batasan tertentu. Untuk lebih jelasnya, lihat pada gambar 2.8.



Gambar 2.8 Ilustrasi Translasi

Persamaan umum dari translasi objek 3 dimensi adalah :

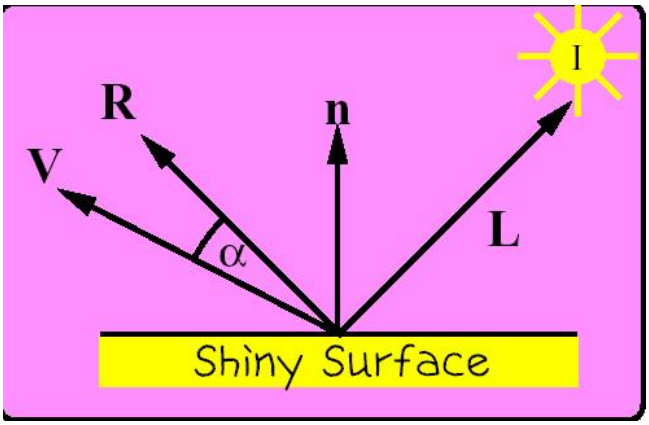
$$P[i]X = P[i]X + \Delta X$$
$$P[i]Y = P[i]Y + \Delta Y \qquad (2\text{-}4)$$
$$P[i]Z = P[i]Z + \Delta Z$$

$P[i]$ adalah titik koordinat awal objek 3 dimensi, sedangkan $\Delta X, \Delta Y, \text{dan } \Delta Z$ adalah jarak geser pada objek.

2.4 Cahaya Speculer

Cahaya Speculer adalah cahaya yang mengacu pada cahaya yang dipantulkan pada permukaan licin dan menciptakan suatu sinar terang dari sumber cahaya yang teriluminasi dan menerangi lingkungan sekitarnya.warna

objek ditentukan oleh radiasi penerangan (iluminasi) bagian-bagian yang diserap serta dipantulkan.seperti pada gambar berikut



$$\text{Specular Component} = K_s * I_p * (\mathbf{R}.\mathbf{V})^n$$

R dan V adalah normal yang dimaksud dengan kuat cahaya dari fungsi cosinus.dengan koefisien secara coba-coba. disini pengamatan dilakukan pada permukaan yang bercahaya, atau terganrung pada sudut pandang dari pengamat. Cahaya yang direfleksikan secara tidak merata pada arah yang berbeda. Reflector yang sempurna (kaca)hanya akan merefleksi kan cahaya pada arah R. Model phong menganggap bahwa ketika a=0 mala, cahaya berkangsung dengan cepat seperti *cos n alpha*.

Bagaimana pabila sumber cahaya bersifat tak hingga? Maka, N dan L bernilai konstan pada permukaan datar. Jika pengamat pada jarak yang tak hingga maka R..V bernilai konstan pada suatu permukaan datar. Pencahayaan maksimum berada pada arah vector setengah, H ketika sejajar dengan n. dimana h=L+V/ | L+V istilah speculer bisa digambarkan sebagai (N+H) dengan sumber cahaya dan pengamat tak terhingga dan H bernilai konstan .

2.5 Skala

Berbeda dengan transformasi geser yang tidak mengubah bentuk objek, transformasi skala akan mengubah bentuk objek sebesat skala Sx dab Sy' , sehingga :

$$(Qx,Qy)= (Px*Sy*Sy)$$

dengan demikian , Matriks transformasi m adalah :

M=

Dan Vektor tr=0.

Transformasi skala dilakukan terhadap titik pusat (0,) larena setiap titil p akam digeser sebesar Sx dari titik pusat dalam sumbu x dan sejauh Sy dalam sumbu Y.

Apabila matrik M berisi nilai negative maka akan diperoleh pencerminan terdahap sumbu v dan Y.jika kedua factor skala berisi nilai yang sama, Sx=Sy=s maka akan diperoleh uniform scaling, dimana objek akan diperbesar pada kedua sumbu besar, jika 0<s<1 maka akandiperoleh objek yang diperkecil (demagnification). Jika factor skala tidak sama , maka akan diperoleh penskalaan defernsial. Jika salah satu dari factor skala sama dengan 1 maka akan diperoleh transfornasi strain.

2.6 Warna

Warna sebenarnya merupakan persepsi kita terhadap pantulan cahaya dari bend-benda didepan mata. Mata manusia dibagi menjadi beberapa bagian, yaitu cornea,iris, lensa, retina dan syaraf mata yang menghubungkan retina dengan otak. Hanya bagian retina ynag berhubungan dengan persepsi kita terhadap warna . sel-sel mata dinagian retina dapat dibagi menjadi beberapa bagian dua macam yaitu :

1. Rods
2. Cones

Sel bertipr rods hanya peka terhadap perbedaan intensitas warna atau dengan kata lain hanya dapat membedakan terang dan gelap, pada umumnya setiap orang mempunyai kurang lebih 150 juta sel. Sel bertipe cones merupakan sel yang peka terhadap spectrum cahaya dengan panjang gelombang dari 390 nm- 720 nm, sel cones sendiri dibagi menjadi 3 macam sel yaitu: Red(L), green Cones (M) dan Blue cobes (S). Sel cones L hanya peka terhadap warna merah dan demikian juga sel cones bertipe S peka terhadap biru. Orang yang seluruh conesnya tudak aktip hanya dapat mempersepsikan terang dan gelap atau hanya dapat ,elihat warna bitam,putih seta area antara hitam dan putih.

2.6.1 Representasi Warna

Ada beberapa model yang digunakan untuk menyatakan warna dasar serta rentang warna yang dihasilkan spertyi RGB, CMY, HSV dan sebagainya

2.6.1.1 Model RGB

Model ini didasarkan berdasarkan pada teori dimana mata manusia peka terhadap panjang gelombang 630 nm(merah), 530 nm (hijau) , dan 450 (biru).dengan mencampur tiga warna dasar tersebut maka kita akan dapat menghasilkan warna yang lain, warna yang dihasilkan dari campuran warna dasar tersebut disebut *additive color* .warna additive terjadi karena sumber cahaya memancarkan sejumlah energi pada panjang gelombang tertentu dan penjumlahan energi pada berbagai panjang gelombang yang dipancarkan oleh sumber cahaya akan menentukan warna akhir yang dihasilkan, atau dapat dirumuskan sebagai

$$W=R+G+B$$

Dengan bilai abtara 0 dan 1, dimana nilai 0 dinyatakan tidak ada komponen tersebut yang digunakan dan nilai 1 menyatakan penakaian penuh dari komponen.

2.6.1.2 Model CMY

Berbeda dengan model RGB yang menghasilkan warna aditif, model CMY menghasilkan warna dengan membuang warna tertentu atau disebut warna substraktif. Warna ini terjadi ketika cahaya putih dilewatkan ke bahan tertentu maka sebagian panjang gelombang diserap oleh bahan, sebagai contoh, warna yang anda lihat pada kertas putih yang diberi tinta biru sebenarnya terjadi karena bahan tinta menyerap panjang gelombang biru dan memantulkan panjang gelombang yang lain.

Warna ini menggunakan tiga warna primer yaitu ; Cyan, Magenta dan Yellow. Warna akhir diperoleh dengan membuang warna primer tersebut sejumlah tertentu.

$$(r,g,b)=(1,1,1)-(c,m,y)CMY$$

Warna biru akan diperolah apabila c= 1, m=1 dan y=0 atau secara praktek kita mencampurkan tinta berwarna cyan dan magenta dengan jumlah yang sama..

2.6.1.3 Model HSV

Model HSV mempresentasikan warna kedalam koordinat yang disusun dari tiga komponen, yaitu : hue, saturation, value. Hue menyatakan warna dominant dan dinyatakan dengan sudut (angle) Value menyatakan intemsitas warna dengan nilai dari 0 sampai 1 sehingga value =0 menyatakan warna hitam

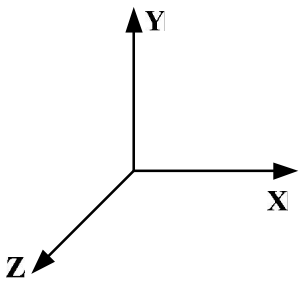
dan value=1 menyatakan warna putih. Saturation menyatakan banyaknya campuran warna dominant dengan warna putih (purity).

2.7 Vektor Grafika

2.7.1 Sistem Koordinat

Salah satu sistem koordinat yang dikenal adalah sistem koordinat kartesian yang digunakan untuk membedakan lokasi atau posisi sembarang titik atau objek dengan titik atau objek yang lain. Sistem koordinat kartesian terdiri atas sistem koordinat kartesian 2 dimensi dan sistem koordinat kartesian 3 dimensi.

Lokasi setiap titik dalam sistem koordinat kartesian 2 dimensi ditentukan oleh dua besaran. Nilai dua besaran tersebut apabila digambar akan membentuk suatu sumbu koordinat mendatar yang sering dikenal dengan sumbu X dan sumbu koordinat tegak yang sering dikenal dengan sumbu Y. Sedangkan dalam sistem koordinat 3 dimensi terdapat satu sumbu lain selain sumbu X dan sumbu Y, yaitu sumbu Z yang arahnya tegak lurus terhadap sumbu X dan sumbu Y. untuk lebih jelasnya, lihat pada gambar 2.9.

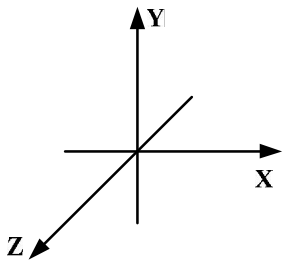


Gambar 2.9 Sistem koordinat kartesian 3 dimensi

Dalam sistem koordinat kartesian 3 dimensi dikenal dua buah aturan, yaitu sistem koordinat katresian 3 dimensi aturan tangan kanan dan sistem koordinat

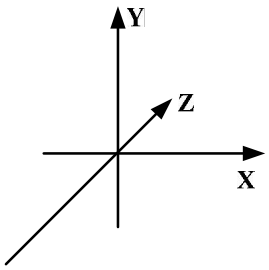
kartesian 3 dimensi aturan tangan kiri, yang selanjutnya lebih dikenal dengan sistem koordinat tangan kanan dan sistem koordinat tangan kiri.

Sistem koordinat tangan kanan dapat diilustrasikan dengan tiga buah jari tangan kanan yaitu ibu jari, jari telunjuk, dan jari tengah dengan telapak tangan kanan menghadap ke pengamat. Ibu jari dianggap sebagai sumbu x, jari telunjuk sebagai sumbu y, dan jari tengah (yang mengarah ke pengamat) sebagai sumbu z. Untuk lebih jelasnya, lihat pada gambar 2.10.



Gambar 2.10 Sistem koordinat kartesian 3 dimensi aturan tangan kanan

Sistem koordinat tangan kiri dapat diilustrasikan dengan tiga buah jari tangan kiri, dengan punggung telapak tangan kiri menghadap pengamat. Dalam hal ini ibu jari sebagai sumbu x, jari telunjuk sebagai sumbu y, dan jari tengah (yang menjauhi pengamat)sebagai sumbu z. Untuk lebih jelasnya, lihat pada gambar 2.11.

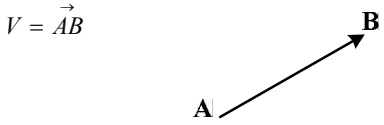


Gambar 2.11 Sistem koordinat kartesian 3 dimensi aturan tangan kiri

2.7.2 Vektor

Vektor adalah besaran fisis yang mempunyai panjang (nilai) dan arah. Vektor-vektor dapat dinyatakan secara geometris sebagai segmen-segmen garis terarah atau panah-panah diruang 2 dimensi atau diruang 3 dimensi. Arah panah menentukan arah vektor dan panjang panah menyatakan besarnya, ekor panah ditentukan titik awal dari vektor dan ujung panah dinamakan titik terminal.

Lambang vektor dinyatakan dengan huruf-huruf kecil misalnya v, a, b, k, w, dan lain-lain. Jika terdapat titik awal vektor V adalah A dan titik terminalnya adalah B, maka akan dituliskan dan digambarkan seperti pada gambar 2.11 berikut



Gambar 2.12 Vektor AB

Vektor dalam dunia 2 dimensi mempunyai 2 buah komponen yaitu i dan j, i adalah komponen arah horizontal (sumbu x) sedangkan j adalah komponen arah vertikal (sumbu y). Dalam dunia 3 dimensi, vektor memiliki 3 komponen yaitu i, j, dan k. Dimana k adalah komponen arah kedalaman (sumbu z), untuk komponen i dan j sama seperti dalam vector 2 dimensi.

2.7.3 Magnitude Vektor

Magnitude vektor adalah besar atau panjang dari vektor. Dapat juga diartikan sebagai panjang garis yang mewakili vektor tersebut. Berdasarkan rumus phytagoras, magnitude atau panjang sebuah vektor dapat dinyatakan sebagai berikut :

$$\|V\| = \sqrt{i^2 + j^2} \text{ di ruang 2 dimensi} \quad (2.5)$$

$$\|V\| = \sqrt{i^2 + j^2 + k^2} \text{ di ruang 3 dimensi} \quad (2.6)$$

Keterangan :

i : komponen arah horizontal (sumbu x)

j : komponen arah vertical (sumbu y)

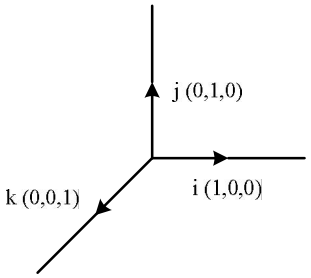
k : komponen arah kedalaman (sumbu z)

2.7.4 Penambahan dan Pengurangan Vektor

Vektor satuan adalah vektor yang mempunyai panjang 1 dan terletak sepanjang sumbu koordinat.

$$\begin{aligned} a - b &= (2i + j + k) - (3i + 2j + k) \\ &= (2-3)i + (1-2)j + (1-1)k \\ &= -i - j \end{aligned} \quad (2.7)$$

Dengan gambar seperti pada gambar 2.13 berikut :



Gambar 2.13 Vektor satuan

Untuk penjumlahan dan pengurangan antara dua buah vektor dapat dilakukan dengan cara menambahkan atau mengurangkan nilai pada masing-masing komponennya.

Misal :

$a = 2i + j + k \qquad b = 3i + 2j + k$

$$a + b = (2i + j + k) + (3i + 2j + k)$$

$$= (2+3)i + (1+2)j + (1+1)k$$

$$= 5i + 3j + 2k \qquad (2.8)$$

Dimana i menyatakan besar dan arah dari koordinat x, j menyatakan besar dan arah koordinat y dan k menyatakan besar dan arah dari koordinat z.

2.7.5Perkalian Titik (*Dot Product*)

Perkalian titik (*dot product*) dapat dilakukan dengan cara mengalikan masing-masing komponennya dari dua buah vektor. Hasil yang diperoleh pada perkalian titik dari dua vektor adalah skalar (besaran yang mempunyai nilai). Aturan yang dipakai dalam perkalian titik adalah sebagai berikut :

$$\begin{aligned} i \cdot j &= 0 \qquad & i \cdot i &= 1 \\ i \cdot k &= 0 \qquad & j \cdot j &= 1 \\ j \cdot k &= 0 \qquad & k \cdot k &= 1 \end{aligned} \tag{2.9}$$

Misal : $a = 2i + j - k$

$$\begin{aligned} a \cdot b &= (2i + j - k) \cdot (i + 2j + 2k) \\ &= 2i \cdot i + j \cdot 2j + (-1)k \cdot 2k \\ &= 2 + 2 - 2 = 2 \end{aligned}$$

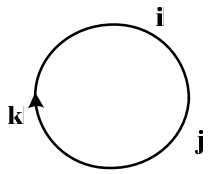
2.7.6 Perkalian Silang (*Cross Product*)

Perkalian silang dapat dilakukan dengan mengalikan dua buah vektor untuk memperoleh sebuah vektor baru yang tegak lurus dengan kedua vektor tersebut. Vektor ini biasanya disebut dengan normal vektor. Aturan yang dipakai dalam perkalian silang adalah sebagai berikut :

$$i \times i = j \times j = k \times k = 0 \tag{2.10}$$

$$\begin{aligned} i \times j &= k \qquad & j \times i &= -k \\ j \times k &= i \qquad & k \times j &= -i \\ k \times i &= j \qquad & i \times k &= -j \end{aligned} \tag{2.11}$$

Dengan gambar seperti pada gambar 2.14 berikut :



Gambar 2.14 Diagram perkalian silang

Berdasarkan gambar diatas, maka hasil perkalian slang dua vektor yang berurutan dalam arah perputaran jarum jam adalah positif dari vektor berikutnya, dan hasil perkalian silang dua vektor yang berurutan dalam arah berlawanan dengan arah perputaran jarum jam adalah negatif dari vektor berikutnya. Misal :

$$a = 2i + j - k \qquad b = i + 2j + 2k$$

$$a \times b = (2i + j - k) \times (i + 2j + 2k)$$

$$= \begin{vmatrix} i & j & k \\ 2 & 1 & -1 \\ 1 & 2 & 2 \end{vmatrix} = \left(\begin{vmatrix} 1 & -1 \\ 2 & 2 \end{vmatrix} i, \quad - \begin{vmatrix} 2 & -1 \\ 1 & 2 \end{vmatrix} j, \quad \begin{vmatrix} 2 & 1 \\ 1 & 2 \end{vmatrix} k \right)$$

$$= (4,-5,3) = 4i - 5j + 3k$$

Dimana i menyatakan besar dan arah dari koordinat x, j menyatakan besar dan arah dari koordinat y, dan k menyatakan besar dan arah dari koordinat z.

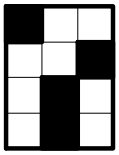
2.8 Matriks

Matriks adalah susunan sekelompok elemen dalam bentuk segiempat siku-siku yang diatur menurut baris dan kolom. Istilah-istilah yang sering digunakan dalam matriks adalah :

1. Baris matriks, yaitu susunan elemen-elemen yang letaknya mendatar (horizontal) dalam matriks tersebut.
2. Kolom matriks, yaitu susunan elemen-elemen yang letaknya tegak (vertikal) dalam matriks tersebut.
3. Elemen matriks, yaitu bagian terkecil dari matriks yang menyusun matriks tersebut.

Banyaknya baris dan kolom suatu matriks menyatakan ukuran matriks tersebut. Ukuran matriks bermacam-macam besarnya, jika suatu matriks terdapat *m* baris dan *n* kolom maka ukuran matriks tersebut adalah *m*x*n*.

Penerapan paling sederhana dari matriks dalam grafika komputer adalah penyimpanan informasi warna pada suatu citra. Misalnya terdapat sebuah citra dengan dimensi 3X4 *pixel*, maka informasi warna disimpan dalam sebuah matriks berukuran 4X3. Untuk lebih jelasnya, lihat pada gambar 2.15.



(a)

$$M = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

(b)

Gambar 2.15 (a) Citra berukuran 3X4 *pixel*, (b) Contoh penerapan matriks berukuran 4X3 untuk menyimpan informasi warna pada citra (a)

2.9 Rendering

Rendering merupakan salah satu teknik pencitraan objek 3 dimensi untuk mendapatkan *image* yang realistis dengan penambahan beberapa efek, seperti pencahayaan dan bayangan. Kebanyakan manusia menggemari sesuatu yang hidup, bukan yang kaku dan statik.

2.9.1 Teknik Pencahayaan (*Lighting*)

Setiap objek dapat terlihat karena ada cahaya. Cahaya yang dapat dari berbagai arah biasanya dapat diketahui asalnya dari sinar dan bayangan yang ditimbulkan. Karena pengaruh cahaya sangat besar terhadap hasil nyata maka faktor pencahayaan harus diperhitungkan. Tetapi mengingat bahwa grafika komputer adalah model matematika dari kehidupan nyata maka pencahayaan juga harus diubah menjadi model matematika. Model matematika itu harus memenuhi persyaratan sebagai berikut :

1. Menghasilkan efek seperti cahaya sungguhan
2. Dapat dihitung dengan cepat

Model pencahayaan tiga dimensi menyangkut yang realistik menyangkut dua elemen penting yang sangat berkaitan erat dengan *shading model*, yaitu :

1. Keakuratan dalam menggambarkan objek.
2. Teknik pencahayaan yang baik.

Teknik pencahayaan pada grafika komputer sering mengacu pada hokum-hukum fisika yang berhubungan dengan intensitas cahaya permukaan. Untuk menyederhanakan perhitungan empiris yang didasarkan pada kalkulasi *photometric* yang disederhanakan. Contoh *ratiosity algoritma* dimana perhitungan intensitas cahaya dengan mempertimbangkan jarak permukaan objek dengan sumber cahaya didalam sebuah *scene*.

Suatu objek bercahaya sedangkan objek itu bukan sumber cahaya, artinya cahaya yang terlihat pada permukaan objek tersebut adalah cahaya pantulan. Total cahaya yang dipantulkan adalah penambahan di kontribusi sumber cahaya dan

pantulan cahaya dari permukaan objek lain dalam *scene*. Artinya suatu permukaan objek tidak sama dengan secara langsung disorot akan diberi penerangan oleh sumber cahaya, akan tetapi terlihat jika objek yang letaknya berdekatan dapat penerangan.

Suatu objek yang punya permukaan kasar maka cahaya yang dipantulkan cenderung akan menyebar ke segala arah, cahaya yang menyebar ini disebut *diffuse reflection*. Apabila sumber cahaya menciptakan cahaya yang terang akan sebuah titik terang disebut *specular*. Efek dari cahaya terang ini lebih ditekankan pada permukaan yang berkilauan daripada permukaan yang tumpul.

Besar sudut yang dihasilkan *specular reflection* terhadap vektor normal. permukaan sama dengan sudut dimana cahaya masuk terhadap vektor normal. permukaan tapi arahnya berlawanan.Teknik pencahayaan diantaranya:

2.9.1.1Cahaya Tersebar

Suatu objek yang mempunyai permukaan yang kasar maka cahaya yang dipantulkan cenderung akan menyebar ke segala arah, cahaya yang menyebar ini disebut cahaya tersebar. Beberapa cahaya menembus permukaan dan diradiasi kembali secara seragam ke dalam semua arah.

Penghitungan cahaya tersebar menggunakan m, v dan s.

1. Sebagaimana cahaya tersebar disebarkan secara seragam dalam semua arah, lokasi mata, v, tidak penting kecuali kalau $v.m < 0$ jika diinginkan intensitas cahaya $I = 0$
2. Hubungan antara kecerahan permukaan dan orientasinya cahaya didasarkan pada $\cos(\theta)$.

$$I_d = I_s P_d \cos(\theta) atau$$
$$I_d = I_s P_d (s.m/|s||m|) \quad (2.12)$$

Keterangan :

$I_d =$ kuat cahaya tersebar (*diffuse*)

$I_s =$ kuat cahaya di sumber cahaya

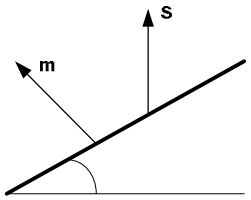
$p_d =$ koefisien pantulan tersebar (*diffuse*)

m = vector normal ke permukaan pada titik P

s = vector dari titik P ke cahaya

v = vector dari titik P ke viewer

Untuk lebih jelasnya, lihat pada gambar 2.16.



Gambar 2.16 Cahaya Pantulan Tersebar (*Diffuse*)

2.9.2 Metode *Shading*

Bayangan (*shading*) adalah bidang yang terbentuk akibat hilangnya sebuah sinar oleh objek yang tidak bisa ditembus oleh sinar tersebut. Metode permukaan tersembunyi dapat digunakan untuk menempuh area dimana bayangan berada yang dihasilkan dari pencahayaan. Pada saat ditentukan letak sebuah bayangan dengan pencahayaan dari arah mana saja, maka bayangan dapat saja terlihat mengikuti bentuk suatu pola permukaan objek lain.

Metode yang digunakan untuk membentuk suatu bayangan antara lain :

2.9.2.1Metode *Flat Shading*

Flat shading adalah metode yang mudah dan cepat untuk membuat bayangan dengan permukaan poligon. Pada metode ini sebuah intensitas tunggal dihitung untuk masing-masing poligon, semua titik pada permukaan poligon dipaparkan dengan nilai intensitas yang sama.

Karakteristik *flat shading* diantaranya :

1. Pemberian *tone* yang sama untuk setiap poligon
2. Penghitungan jumlah cahaya mulai dari titik tunggal pada permukaan.
3. Penggunaan satu normal untuk seluruhnya.

Secara umum *flat shading* dapat menghasilkan bayangan yang akurat dengan ketentuan sebagi berikut :

1. Objek berbentuk polihendra, yaitu jaring yang mempunyai ruang terhingga dan tertutup.
2. Semua sumber cahaya jauh dari permukaan objek.
3. Posisi penglihatan yang cukup jauh dari permukaan.

2.9.2.2Metode *Guround Shading*

Metode ini merender sebuah permukaan poligon dengan interpolasi linier yaitu nilai intensitas yang mengenai setiap permukaan berbeda. Warna yang dipantulkan dihitung tiap *vertex* (garis) kemudian secara halus diinterpolasikan. Karakteristik bayangan yang dihasilkan :

1. Bayangan yang dihasilkan halus (tampak nyata)

2. Penggunaan level abu-abu yang berbeda disepanjang poligon diinterpolasikan di antara titik.

Cara untuk menghasilkan bayangan dengan menggunakan metode ini adalah :

1. Tentukan satuan vektor normal rata-rata pada setiap titik ujung poligon.
2. Pakaikan model iluminasi untuk setiap titik untuk menghitung intensitas titik.
3. Interpolasikan secara linier intensitas titik pada permukaan poligon.

2.10 OpenGL

OpenGL adalah suatu *software* penghubung ke *hardware* grafik. Alat penghubung ini terdiri dari ± 120 perintah berbeda, yang digunakan untuk menetapkan operasi pada objek sehingga menghasilkan aplikasi 3 dimensi yang interaktif. OpenGL dirancang untuk bekerja secara efisien sekalipun grafik komputer yang digunakan bukan grafik komputer yang menunjang program-program grafik lainnya.

OpenGL mempunyai sebuah perkongsian bebas yang dikenal dengan nama OpenGL Architecture Review Board (ARB). ARB dibentuk pada tahun 1992 dan berperan sebagai pengendali perancangan-perancangan OpenGL secara terperinci sehingga dapat mendukung grafik dua dimensi dan tiga dimensi kedalam Application Programming Interface (API), OpenGL dapat diimplementasikan didalam bahasa Visual basic, C++, dll.

Setiap aplikasi visual yang menuntut kemampuan animasi 3 dimensi secara maksimum maka akan menghasilkan simulasi yang berkualitas tinggi dan OpenGL memiliki kemampuan tersebut bahkan OpenGL memilki kelbihan tersendiri disbanding dengan aplikasi-aplikasi lain yang sejenis, diantaranya:

1. Stabil

   Implementasi OpenGL telah dirancang selama bertahun-tahun dan dengan platform yang luas. Setiap penambahan atau pembaharuan perancangan-perancangan yang dikendalikan ARB, selalu diumumkan kepada para pengembang untuk mengadakan pembaharuan juga sehingga dapat dipastikan aplikasi yang berjalan tidak pernah usang.

2. Mudah untuk menyesuaikan

   Semua aplikasi OpenGL menghasilkan tampilan visual yang konsisten di *Application Programming Interface* (API) manapun OpenGL digunakan, dengan mengabaikan *operating system* yang ada.

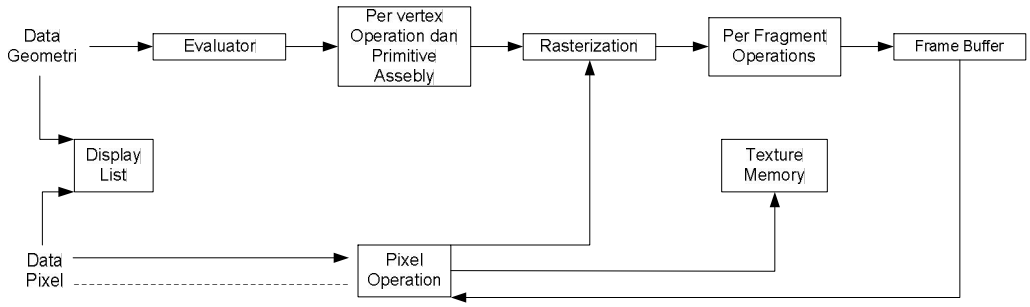
3. Mudah Penggunaannya

   OpenGL mempunyai susunan desain yang intuitif, perintah yang logis dan rutin yang khas sehingga memudahkan penggunaannya untuk mengahsilkan sebuah paket grafik yang berkualitas.

4. Organisasi

   Sebuah perkongsian yang berdiri bebas. ARB yang selalu mengendalikan perkembangan OpenGL dalam mendukung industri luas. OpenGL adalah satu-satunya yang benar-benar terbuka, netral dan multiplatform standar grafik.

Perkembangan OpenGL yang pesat dan inovatif diterapkan terhadap cara-cara menggabungkan objek-objek, pemetaan tekstur, efek-efek khusus dan beberapa fungsi visual.Untuk lebih jelasnya, lihat pada gambar 2.18.



Gambar 2.18 Diagram Alur OpenGL

Diagram blok diatas memberikan suatu gambaran abstrak bagaimana OpenGL memproses data. Didalam diagram, perintah masuk dari kiri dan melalui berbagai proses, perintah tersebut akan diingat dan dijabarkan. Beberapa perintah menetapkan objek geometris untuk digambar dan yang lain mengendalikan bagaimana objek ditangani sepanjang langkah-langkah proses berlangsung.

2.10.1Perintah dalam OpenGL

Sintak-sintak perintah dalam OpenGL biasanya menggunakan perfik ‘gl’ yang diikuti dengan nama sintak, dimana masing-masing kata diawali dengan huruf besar, misalnya GL_COLOR_BUFFER_BIT, GL_CURRENT_COLOR, GL_EDGE_FLAG, dan lain-lain.

Beberapa sintak perintah dalam OpenGL diakhiri dengan format-format ter tentu, misalnya 3f, 4d, 4ub, dan lain-lain. Digit didepan huruf terakhir

menyatakan jumlah parameter yang harus disertakan dalam nama sintak yang diikutinya, misalnya gl color 4f(par_1, par_2, par_3, par_4),glvertex3f(par_1, par_2, par_3, par_4), dan lain-lain. Huruf terakhir pada sintak OpenGL menyatakan tipe parameternya. Table 2.1 menunjukkan macam-macam contoh perintah-perintah dalam OpenGL.

Table 2.1 Contoh perintah-perintah dalam OpenGL

Perintah	**Arti**	**Keterangan**
glVertex2i(x,y);	Lokasi titik berada di (x,y)	Tipe argumennya adalah integer dan 2 dimensi, yaitu x dan y.
glVertex2f(x,y);	Lokasi titik berada di (x,y)	Tipe argumennya adalah float dan 2 dimensi, yaitu x dan y.
glVertex3i(x,y,z);	Lokasi titik berada di (x,y,z)	Tipe argumennya adalah integer dan 3 dimensi, yaitu x, y, dan z.
glVertex3f(x,y,z);	Lokasi titik berada di (x,y,z)	Tipe argumennya adalah float dan 3 dimensi, yaitu x, y, dan z.
glClearColour(R,G,B, α);	Warna latar belakang	4 komponen warna yaitu Red, Green, Blue, dan Alpha.
glClearColour3f(R,G,B);	Warna latar muka	3 komponen warna yaitu Red, Green, dan Blue.
glColour4f(R,G,B, α);	Warna latar muka	4 komponen warna yaitu Red, Green, Blue, dan Alpha.
glPointSize(k);	Ukuran titik ke piksel	Besar kecilnya ukuran titik tergantung pada k (integer).
glBegin(GL_POINTS);	Titik	Objek primitif
glBegin(GL_LINES);	Garis	Objek primitif
glBegin(GL_LINE_STRIP);	Poligaris	Objek primitif
glBegin(GL_LINE_LOOP);	Poligaris tertutup (polygon)	Objek primitif
glBegin(GL_TRIANGLES);	Segitiga	Objek primitif
glBegin(GL_TRIANGLE_STRIP);	Segitiga	Objek primitif

glBegin(GL_TRIANGLE_FAN);	Segitiga	Objek primitif
glBegin(GL_QUADS);	Segiempat	Objek primitif
glBegin(GL_QUAD_STRIP);	Segiempat	Objek primitif
glBegin(GL_LINE_STIPPLE);	Garis putus-putus	Objek primitif
glBegin(GL_POLY_STIPPLE);	Polygon dengan pola tertentu	Objek primitif
glBegin(GlintX1,GlintY1,GlintX2,GlintY2);	Segiempat siku-siku	Objek primitive, dan ukuran segiempat ditentukan oleh dua titik yaitu (x_1, y_1) dan (x_2, y_2).
glEnd();	Akhir perintah OpenGL	-
gl Lighting	Cahaya Lokasi tertentu	
gl shading	corak	
gl Proyeksi	Sudut pandang terhadap objek tertentu	