

## BAB II

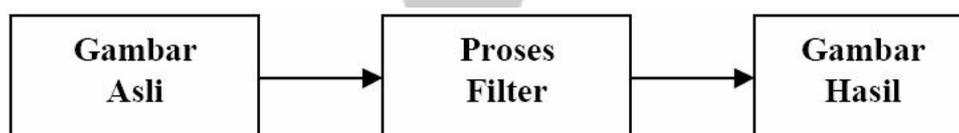
### Tinjauan Pustaka

#### II.1. Pengolahan Citra Digital

Citra yang diperoleh dari lingkungan masih terdiri dari warna yang sangat kompleks sehingga masih diperlukan proses lebih lanjut agar *image* tersebut dapat untuk memproses sebuah sistem.

Pengolahan Citra Digital merupakan suatu metode yang digunakan untuk mengolah atau memproses dari gambar asli sehingga menghasilkan gambar lain yang sesuai dengan kebutuhan. Pengambilan gambar bisa dilakukan oleh kamera video atau alat-alat yang lain yang dapat digunakan untuk mentransfer gambar. Dalam pengolahan citra, dilakukan operasi terhadap citra asli menjadi citra baru berdasarkan citra asli.

Contoh dari suatu proses pengolahan citra digital adalah misalnya terdapat suatu gambar yang terlalu gelap maka dengan proses pengolahan citra digital dengan cara dan metode tertentu maka gambar tersebut bisa diproses sehingga mendapatkan gambar yang jelas. Secara garis besar bisa gambarkan seperti blok diagram pada gambar 2.1 dibawah ini:



Gambar 2.1 Blok Diagram Pengolahan Citra (Munir,2004)

## **II.2. Pengertian Citra**

Suatu citra yang tampil dilayar monitor merupakan kumpulan dari piksel-piksel. Piksel (piksel atau *picture element*) adalah titik di layar monitor yang dapat diatur untuk menampilkan warna tertentu. Piksel disusun di layar monitor dalam susunan baris dan kolom. Setiap piksel yang terletak pada koordinat layar memiliki nilai 0 berarti keadaan piksel mati dan menyatakan warna putih sedangkan nilai 1 berarti keadaan piksel hidup. Susunan piksel dalam baris dan kolom dinamakan resolusi gambar. Resolusi monitor yang sering dijumpai adalah 640 X 480, yang mampu menampilkan piksel sejumlah 640 baris dan 480 kolom, sehingga total yang ditampilkan di layar adalah 307.200 piksel. Melalui piksel inilah suatu gambar dapat dimanipulasi agar sesuai dengan kebutuhan pemakai.

### **II.2.1. Resolusi Citra**

Setiap posisi dalam citra dapat berisi piksel. Ini berarti bahwa tidak ada sesuatu yang mampu menunjukkan bayangan samar dari objek. Piksel gambar yang kecerahannya dibawah tingkat tertentu diwakili oleh "0" sedangkan di atasnya diwakili oleh "1", dengan demikian semua citra di dalam memori komputer dapat diwakili oleh logika "1" dan "0". Sekarang dapat menghitung jumlah memori yang dibutuhkan untuk menyimpan citra tersebut. Sebagai contoh citra yang mewakili resolusi 256x256 piksel ( jumlah total 65536 ), karena setiap piksel diwakili oleh "0" dan "1", maka komputer membutuhkan satu bit untuk menyimpan setiap piksel, sehingga dibutuhkan

total bit sekitar 64 kb, dengan demikian jumlah memori akan bertambah besar jika resolusi citra bertambah.

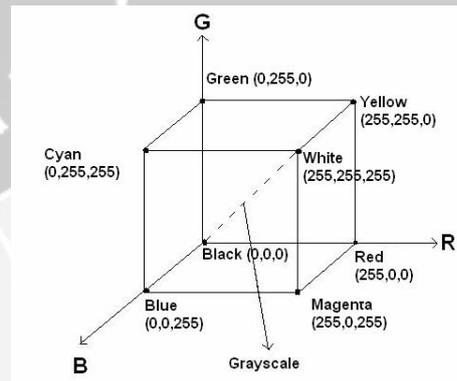
### II.2.2. Warna RGB

Pada mode warna RGB setiap titik pada layar berisi angka yang bukan menunjukkan intensitas warna dari titik tersebut, melainkan menunjukkan intensitas yang dipilih pada suatu tabel. Jadi pada setiap titik, dapat dipilih salah satu warna pada tabel. Adapun masing-masing warna dari tabel tersebut memiliki tiga buah kombinasi angka, yaitu R, G, dan B yang menentukan proporsi warna merah (*Red*), hijau (*Green*), dan biru (*Blue*) dari warna tersebut. R, G, dan B masing-masing memiliki *range* antara 0 hingga 255 sehingga jumlah warna pada tabel yang dapat dipilih untuk mengisi warna pada sebuah titik ialah  $256 \times 256 \times 256 = 16,7$  juta warna.

Sedangkan warna *grayscale* diperoleh dengan cara mengisi seluruh isi tabel warna dengan warna-warna yang memiliki proporsi antara R, G, dan B yang sama. Yaitu mulai dari  $R = 0, G = 0$  dan  $B = 0$  hingga  $R = 255, G = 255,$  dan  $B = 255$ . Sehingga akan diperoleh 256 warna dari warna hitam kemudian perlahan-lahan bertambah terang sampai warna putih terang.

Warna RGB direpresentasikan dalam unit kubus dengan sumbu koordinat R,G,dan B yang saling berpotongan. Gambar 2.1 merupakan model warna RGB dimana warna hitam memiliki nilai RGB  $(0,0,0)$  dan warna putih memiliki nilai RGB  $(255,255,255)$ . Garis dari warna hitam  $(0,0,0)$  ke putih  $(255,255,255)$  adalah garis skala keabuan (*grayscale*).

Untuk lebih jelasnya warna RGB dapat dilihat pada gambar 2.2 berikut :



Gambar 2.2. Model warna RGB (Prosis,1994)

### II.3. File BMP

Untuk melakukan penyimpanan citra, *user* perlu menentukan format file citra (*Image File Format*). Format File Citra merupakan cara untuk menyimpan citra grafis digital. Melakukan suatu penyimpanan citra, berarti sama dengan melakukan penyimpanan suatu array dua dimensi. Nilai pada array dua dimensi ini yakni merupakan representasi data yang masing - masing berhubungan dengan piksel pada citra. Demikian file BMP yang merupakan salah satu dari file citra.

Format BMP terdiri dari beberapa bagian utama yaitu *file header*, *information header*, *color map* dan citra. *File header* ini berfungsi untuk menentukan apakah suatu *file* berformat BMP atau tidak. Penyajiannya secara lengkap dapat dilihat pada Tabel 2.1. dibawah ini.

Tabel 2.1. Windows BITMAP File Header

Offset	Ukuran	Nama	Keterangan
0	2	BfType	ASCII "BM"
2	4	BfSize	Ukuran dalam longword setiap <i>file</i>
6	2	bfReserved1	No1
8	2	bfReserved2	No1
10	4	BfOffbits	Byte offset setelah <i>header</i> dimana citra dimulai

Informasi tentang *information header* dapat dilihat pada Tabel 2.2. sebagai berikut:

Tabel 2-2 Informasi header

Offset	Ukuran	Nama	Keterangan
14	4	bi Size	Ukuran dari header ini 40 <i>byte</i>
18	4	bi Width	Lebar citra
22	4	bi Height	Tinggi citra
26	2	bi Planes	Jumlah bidang citra harus 1
28	2	bi BitCount	Bit perpixel 1,4,8,24
30	4	bi Compression	Tipe kompresi data
34	4	bi SizeImage	Ukuran dari citra yang telah dikompresi

38	4	bi XPelsPermeter	Resolusi horizontal dalam piksel/meter
42	4	bi YPelsPermeter	Resolusi vertikal dalam pixe/meter
46	4	bi ClrUsed	Jumlah warna yang digunakan
50	4	bi Important	Jumlah warna pokok untuk citra
54	4*N	bi Colors	Color map

Untuk *color map* ini masing-masing warna disimpan dengan ukuran 4 *byte* yang masing-masing *byte*-nya berisi informasi tentang warna merah, hijau, biru dan intensitasnya seperti pada Tabel 2.3.

Tabel 2.3. Windows RGBQUAD color mapentry

Offset	Nama	Keterangan
0	RgbBlue	Nilai biru
1	RgbGreen	Nilai hijau
2	RgbRed	Nilai merah
3	rgbReserved	Nol

#### II.4. Filter Warna (*Color Filter*)

*Filtering* adalah suatu proses pengambilan sebagian sinyal dari frekuensi tertentu, dan membuang sinyal pada frekuensi yang lain. *Filtering* pada citra (*image*) juga menggunakan prinsip yang sama, yaitu mengambil fungsi citra pada frekuensi-frekuensi tertentu.

Pada gambar citra wajah terdapat tiga warna dominan yaitu merah, biru, dan kuning. Oleh sebab itu, dilakukan *filtering* terhadap warna-warna tersebut dengan menggunakan filter warna, artinya warna dominan yang diinginkan dipertahankan dan menghilangkan warna yang tidak diinginkan.

### II.5. Kuantisasi Rata-rata

Vektor kuantisasi rata-rata adalah suatu metode kuantisasi vektor dimana sebuah gambar dibagi menjadi bagian-bagian gambar yang lebih kecil (*resampling*) dan nilai pada setiap bagian segmen (*sampling*) diwakili oleh hasil rata-rata terhadap bagian *sampling* tersebut. Sebagai contoh proses kuantisasi rata-rata, digunakan pada gambar masukan (*image query*) dengan ukuran 64x64, bila digunakan ukuran *sampling* (segmen) 4x4 maka akan diperoleh 16x16 bagian *sampling*. Artinya, *image query* tersebut mempunyai ukuran yang lebih kecil, yaitu 16x16.

Pembelajaran dan penentuan ciri dari masing-masing gambar citra wajah dapat dilakukan dengan berbagai model ciri kuantisasi. Kuantisasi rata-rata yang digunakan dalam proses pembentukan ciri dihitung dengan menggunakan persamaan berikut :

$$\bar{x} = \frac{\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} x_{i,j}}{n_i \cdot n_j} \quad (2.1)$$

dimana :

$\bar{x}$  : nilai rata-rata dari setiap bagian *sampling*,

$x_{i,j}$  : nilai pada masing-masing piksel dalam satu bagian *sampling*,

$n_i, n_j$  : masing-masing adalah ukuran baris dan kolom pada tiap bagian *sampling*.

## II.6. Euclidean Distance

Untuk mencari jarak antara piksel menggunakan Jarak Euclidean. Jarak Euclidean dapat dianggap sebagai jarak yang paling pendek antara dua titik dan pada dasarnya sama halnya dengan persamaan *Pythagoras* ketika digunakan didalam dua dimensi. Secara matematis dapat dituliskan didalam persamaan berikut:

$$d(i,j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \dots + |x_{in} - x_{jn}|^2} \quad (2,2)$$

Dimana:

$d(i, j)$  = jarak antara 2 titik

$x_{in}$  = koordinat titik pertama

$x_{jn}$  = koordinat titik kedua

Ketika menggunakan fungsi jarak Euclidean untuk membandingkan jarak, tidak diperlukan untuk menghitung akar dua sebab jarak selalu merupakan angka-angka positif.

Kemudian, dicari nilai selisih terkecil dari citra wajah yang dimasukkan dan citra-citra wajah yang ada dalam database. Selisih terkecil antara kedua citra ini digunakan sebagai hasil pengenalan.