

BAB II

LANDASAN TEORI

2.1 Pengertian Citra

Citra atau *image* adalah representasi spasial dari suatu objek yang sebenarnya dalam bidang dua dimensi yang biasanya ditulis dalam koordinat cartesian x-y, dan setiap koordinat merepresentasikan satu sinyal terkecil dari objek (Kulkarni, 2001). Fungsi citra adalah model matematika yang sering digunakan untuk menganalisis dimana semua fungsi analisis digunakan untuk mempertimbangkan citra sebagai fungsi dengan 2 variabel.

Citra merupakan gambaran yang terekam oleh kamera atau oleh sensor. mengutarakan pengertian tentang citra yaitu:

1. Gambaran obyek yang dibuahkan oleh pantulan atau pembiasan sinar yang difokuskan oleh sebuah lensa atau sebuah cermin.
2. Gambaran rekaman suatu obyek (biasanya berupa gambaran pada foto) yang dibuat dengan cara optik, elektro-optik, optik mekanik atau elektronik. Pada umumnya gambar digunakan bila radiasi elektromagnetik yang dipancarkan atau dipantulkan dari suatu obyek tidak langsung direkam pada film. Sedangkan penginderaan jauh ialah ilmu dan seni untuk memperoleh informasi tentang obyek, daerah, atau gejala dengan jalan menganalisis data yang diperoleh dengan menggunakan alat tanpa kontak langsung terhadap obyek, daerah, atau gejala yang dikaji.

2.2 Pengolahan Citra

Pengolahan citra (*image Processing*) merupakan suatu sistem dimana proses dilakukan dengan masukan berupa citra dan hasilnya juga berupa citra (Basuki, 2005). Pada awalnya pengolahan citra ini dilakukan untuk memperbaiki kualitas citra, namun dengan berkembangnya dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses computer, serta dapat mengambil informasi dari suatu citra, maka image processing tidak dapat dilepaskan dengan bidang computer vision

Sesuai dengan perkembangan computer vision itu sendiri, pengolahan citra mempunyai dua tujuan utama yakni sebagai berikut.

1. Memperbaiki kualitas citra, dimana citra yang dihasilkan dapat menampilkan informasi secara jelas atau dengan kata lain manusia dapat melihat informasi yang diharapkan dengan menginterpretasikan citra yang ada. Dalam hal ini interpretasi terhadap informasi yang ada tetap dilakukan oleh manusia.
2. Mengekstraksi informasi cirri yang menonjol pada suatu citra dimana hasilnya adalah informasi citra dimana manusia mendapatkan informasi cirri dari citra secara numeric atau dengan kata lain computer melakukan interpretasi terhadap informasi yang ada pada citra melalui besaran-besaran data yang dapat dibedakan secara jelas.

2.3 Citra Grayscale

Menurut Basuki (2005), Proses awal yang banyak dilakukan dalam image processing adalah merubah citra berwarna menjadi grayscale.

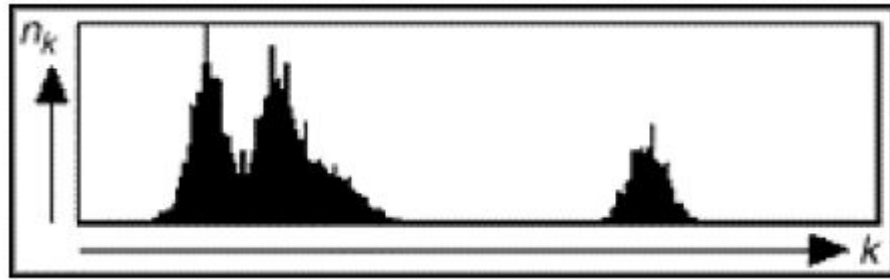
Hal ini digunakan untuk menyederhanakan model citra. Pada citra berwarna terdiri dari 3 layer matrix, yaitu R-layer, G-layer dan B-layer sehingga untuk melakukan proses-proses selanjutnya tetap diperhatikan tiga layer diatas. Bila setiap proses perhitungan dilakukan menggunakan tiga layer, berarti dilakukan tiga perhitungan yang sama. Dengan demikian, konsep itu dirubah dengan mengubah tiga layer diatas menjadi 1 layer matrix grayscale dan hasilnya adalah citra grayscale. Dalam citra ini tidak ada lagi warna, yang ada adalah derajat keabuabuan.

Untuk mengubah citra berwarna yang mempunyai nilai matrix masing-masing r, g, dan b menjadi citra grayscale dengan nilai s, maka konversi dapat dilakukan dengan mengambil rata-rata dari nilai r, g, dan b sehingga dapat dituliskan menjadi :

$$S = (r + g + b) / 3$$

2.4 Histogram

Menurut Sutoyo (2009) Histogram adalah grafik yang menunjukkan frekuensi kemunculan setiap nilai gradasi warna. Misalkan citra digital memiliki L derajat keabuan, yaitu dari nilai 0 sampai $L - 1$ (misalnya pada citra dengan kuantisasi derajat keabuan 8-bit, nilai derajat keabuan dari 0 sampai 255). Gambar 2.1 memperlihatkan contoh sebuah histogram citra, yang dalam hal ini k menyatakan derajat keabuan dan n_k menyatakan jumlah *pixel* yang memiliki nilai keabuan k .



Gambar 2.1 Histogram Citra (Sutoyo, 2009)

Seringkali pada beberapa operasi pengolahan citra jumlah *pixel* yang memiliki derajat keabuan k dinormalkan terhadap jumlah seluruh *pixel* di dalam citra,

$$h_k = \frac{n_k}{n}, \quad k = 0, 1, \dots, L-1$$

Sehingga $0 \leq h_i \leq 1$. Persamaan 1 di atas menyatakan frekuensi kemunculan nisbi (*relative*) dari derajat keabuan pada citra tersebut. Khusus untuk citra berwarna, histogramnya dibuat untuk setiap kanal *RGB* (merah, hijau, dan biru). Histogram citra menunjukkan banyak hal tentang kecerahan (*brightness*) dan kontras (*contrast*) dari sebuah gambar. Puncak histogram menunjukkan intensitas *pixel* yang menonjol. Lebar dari puncak menunjukkan rentang kontras dari gambar. Citra yang mempunyai kontras terlalu terang (*overexposed*) atau terlalu gelap (*underexposed*) memiliki histogram yang sempit. Histogramnya terlihat hanya menggunakan setengah dari daerah derajat keabuan. Citra yang baik memiliki histogram yang mengisi daerah derajat keabuan secara penuh dengan distribusi yang merata pada setiap derajat keabuan *pixel*. Histogram adalah alat bantu yang berharga dalam pekerjaan pengolahan citra baik secara kualitatif maupun kuantitatif. Histogram berguna antara lain untuk perbaikan kontras dengan teknik histogram equalization dan memilih nilai ambang untuk melakukan segmentasi objek.

2.5 Kecerahan (*Brightness*)

Menurut Sutoyo (2009) Sebuah citra grayscale 256 warna akan tampak gelap bila seluruh komponen warnanya berada mendekati 0, sebaliknya citra akan tampak terang bila seluruh komponen warnanya mendekati 256, untuk mengontrol nilai warna citra agar diperoleh tingkat kecerahan sesuai yang diinginkan maka fungsi kecerahaan berikut digunakan.

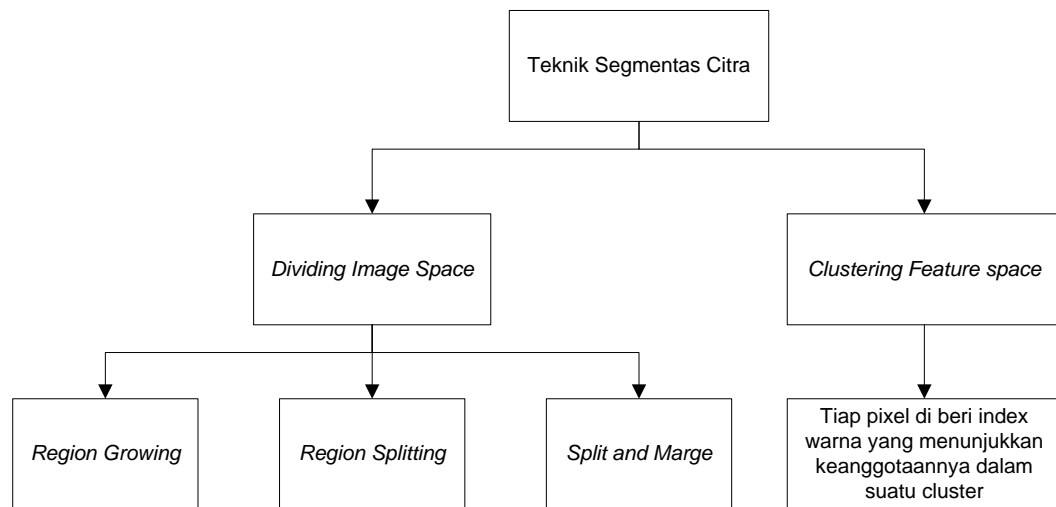
$$f_o(x,y) = f_i(x,y) + k \dots\dots\dots$$

dimana f_o adalah nilai intensitas (warna) setelah dilakukan operasi fungsi kecerahaan f_i adalah nilai sebelum dilakukan operasi fungsi kecerahaan, dan k adalah konstanta kecerahaan. Bila nilai k positif maka citra hasil operasi lebih cerah disbanding dengan citra asli. Sebaliknya, bila nilai k negatif maka citra hasil operasi lebih redup disbanding dengan citra asli. Untuk citra true color, fungsi tersebut diterapkan untuk masing-masing warna *Red*, *Blue* dan *Green*

2.6 Pengertian Segmentasi Citra

Segmentasi citra adalah suatu proses membagi suatu citra menjadi wilayah-wilayah yang homogen (Jain, 1989). Menurut Jain (1989), segmentasi citra dapat dibagi dalam beberapa jenis, yaitu *dividing image space* dan *clustering feature space*. Jenis yang pertama adalah teknik segmentasi dengan membagi *image* menjadi beberapa bagian untuk mengetahui batasannya, sedangkan teknik yang kedua dilakukan dengan cara memberi index warna pada tiap piksel yang menunjukkan keanggotaan dalam suatu segmentasi. Teknik segmentasi citra,

menurut Jain(1989), dapat dilihat pada diagram berikut.



Gambar 2.2 Teknik segmentasi citra (Jain,1989)

Adapun teknik segmentasi tersebut dapat dilakukan dengan beberapa pendekatan sebagai berikut :

1. Pendekatan *Edge-Based*

Pendekatan ini melakukan proses deteksi sisi dengan operator gradient. Masukannya berupa citra *gray level* dan keluarannya berupa citra *edge* (biner). Selanjutnya dilakukan proses *region growing* dengan masukan citra asli (*gray-level*) dan citra *edge*. Proses pembentukan suatu wilayah akan berhenti bila menjumpai piksel *edge*. Kekurangan dari pendekatan ini adalah belum tentu menghasilkan *edge* yang kontinu, mengakibatkan terjadinya kebocoran wilayah (wilayah-wilayah yang tidak tertutup).

2. Pendekatan *Region-Based*

Pendekatan ini memerlukan *criteria of uniformity*, memerlukan penyebaran *seeds* atau dapat juga dengan pendekatan *scan line*, kemudian dilakukan proses *region growing*. Kekurangan dari pendekatan ini adalah belum tentu menghasilkan wilayah-wilayah yang bersambungan.

3. Pendekatan *Hybrid*

Pendekatan ini melakukan proses deteksi sisi untuk menghasilkan citra sisi (piksel *edge* dan piksel *non-edge*), melakukan pemisahan wilayah dengan metode *connected region*. (*Connected regions* adalah set piksel 4-tetangga yang bukan piksel *edge*), dan selanjutnya dilakukan proses merging regions. Pendekatan ini bertujuan untuk mendapatkan hasil segmentasi dengan wilayah-wilayah yang tertutup dan bersambungan.

Sedangkan menurut Sutoyo (2009) segmentasi citra bertujuan untuk membagi wilayah-wilayah yang homogen. Segmentasi adalah salah satu metode penting yang digunakan untuk mengubah citra input ke dalam citra output berdasarkan atribut yang diambil dari citra tersebut. Segmentasi membagi citra ke dalam daerah intensitasnya masing-masing sehingga bisa membedakan antara objek dan backgroundnya. Pembagian ini tergantung pada masalah yang akan diselesaikan. Segmentasi harus dihentikan apabila masing-masing objek telah terisolasi atau terlihat dengan jelas. Tingkat keakurasian segmentasi tergantung pada tingkat keberhasilan prosedur analisis yang dilakukan dan diharapkan proses segmentasi memiliki tingkat keakuratan yang tinggi. Algoritma dari segmentasi terbagi dalam dua macam yaitu :

1. Diskontinuitas

Pembagian citra berdasarkan perbedaan dalam intensitasnya.

Contohnya titik, garis, dan tepi

2. Similaritas

Pembagian citra berdasarkan kesamaan-kesamaan kriteria yang

dimilikinya, contohnya *thresholding*, *region growing*, *region splitting*, dan *region merging*.

2.7 Metode Thresholding

Menurut Sutoyo (2009) Metode ini menggunakan nilai ambang T sebagai patokan untuk memutuskan sebuah pixel diubah menjadi hitam atau putih.

Biasanya dihitung dengan persamaan : $T = (f_{maks} + f_{min}) / 2$

Dimana f_{max} adalah nilai intensitas maksimum pada citra dan f_{min} adalah intensitas minimum pada citra. Jika $f(x,y)$ adalah nilai intensitas pixel pada posisi (x,y) maka pixel tersebut diganti putih atau hitam tergantung kondisi berikut.

$$F(x,y) = 255, \text{ jika } f(x,y) \geq T$$

$$F(x,y) = 0, \text{ jika } f(x,y) < T$$

Sebagai contoh, misalkan diketahui citra grayscale 4 x 4 piksel dengan kedalaman 8 bit seperti berikut :

200	230	150	75
240	50	170	90
210	100	120	80
100	90	200	230

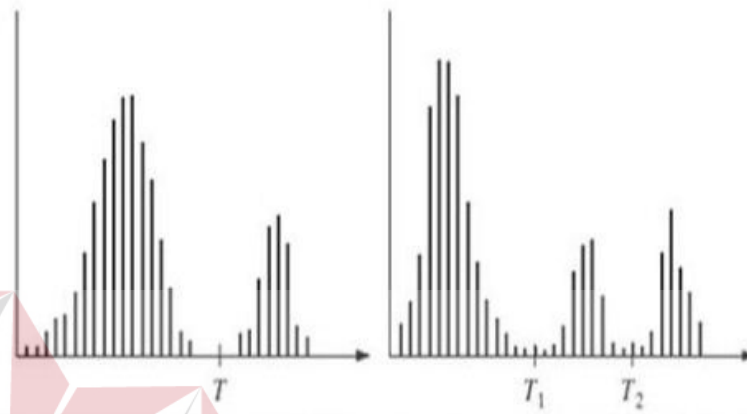
Dengan metode ini, nilai *threshold* T adalah :

$$T = (f_{max} + f_{min}) / 2 = (240 + 50) / 2 = 145$$

Bila nilai $T = 145$ maka diterapkan untuk citra tersebut diperoleh sebagai berikut .

255	255	255	0
255	0	255	0
255	0	0	0
0	0	255	255

Menurut Sutoyo (2009) Bila histogram citra terbagi menjadi dua wilayah seperti pada gambar maka global thresholding bisa digunakan untuk mendapatkan nilai *thresholding* T yang tepat sehingga bagian objek dan latar belakang citra bisa ditentukan .



Gambar 2.3 *Global Thresholding* (Sutoyo, 2009)

Dengan metode ini, threshold dimodifikasi secara berulang-ulang sampai ditemukan nilai yang cocok . langkah-langkah dalam menentukan *threshold* T adalah sebagai berikut.

1. Pilih nilai T awal, yaitu nilai rata-rata dari intensitas citra.
2. Bagi citra menjadi dua daerah, misalnya R_1 dan R_2 , menggunakan nilai T awal yang telah ditentukan.
3. Hitung nilai rata-rata intensitas μ_1 dan μ_2 masing-masing untuk daerah R_1 dan daerah R_2
4. Hitung nilai *threshold* yang baru dengan rumus $T = \frac{\mu_1 + \mu_2}{2}$
5. Ulangi langkah 2 sampai 4 hingga nilai-nilai μ_1 dan μ_2 tidak berubah lagi.

Saat itulah nilai T merupakan nilai yang dicari.

2.8 Pengenalan Pola

Pengenalan pola (*pattern recognition*) adalah suatu ilmu untuk mengklarifikasi atau menggambarkan sesuatu berdasarkan pengukuran kuantitatif fitur (ciri) atau sifat utama dari suatu objek (Putra, 2010). Pola sendiri adalah suatu entitas yang terdefinisi dan dapat diidentifikasi serta diberi nama. Sidik jari adalah satu contoh pola. Pola bias merupakan kumpulan hasil pengukuran atau pemantauan dan bias dinyatakan dalam notasi vector atau metriks.

2.9 Barcode

Menurut Malik (2010) , *barcode* merupakan sejenis kode yang mewakili data atau informasi tertentu, biasanya jenis dan harga barang , contohnya seperti untuk makanan dan buku. *Barcode* berbentuk batangan balok dan berwarna hitam putih ini mengandung satu kumpulan kombinasi batang yang berlainan ukuran yang disusun sedemikian rupa.

Menurut Malik (2010) *barcode* memiliki banyak manfaat yang beraneka ragam, diantaranya :

1. Akurasi

Barcode mampu meningkatkan akurasi dengan mengurangi kesalahan manusia ketika melakukan pemasukan data secara manual atau barang yang salah baca.

2. Kemudahan pemakaian

Dengan perangkat keras dan perangkat lunak yang tepat, *barcode* bisa memaksimalkan proses otomatisasi pengumpulan data.

3. Keseragaman pengumpulan data

Beragam standar pemenuhan dan simbologi barcode yang terstandarisasi menjamin informasi diterima dan disampaikan dengan cara yang benar sehingga dapat diterima dan dipahami secara umum.

4. Umpan balik yang tepat waktu

Barcode menawarkan umpan balik yang tepat waktu. Begitu muncul, data bisa diterima dengan cepat sehingga memungkinkan pengambilan keputusan yang cepat berdasarkan informasi terbaru.

5. Keamanan

Dengan menggunakan *barcode*, perusahaan mampu memperkecil kesalahan yang terjadi. Misalnya penukaran harga label produk yang dilakukan konsumen di suatu supermarket.

6. Meningkatkan produktifitas

Barcode mampu membuat aktivitas operasional dalam bisnis menjadi lebih cepat sehingga produktivitas dapat meningkat.

7. Meningkatkan profit

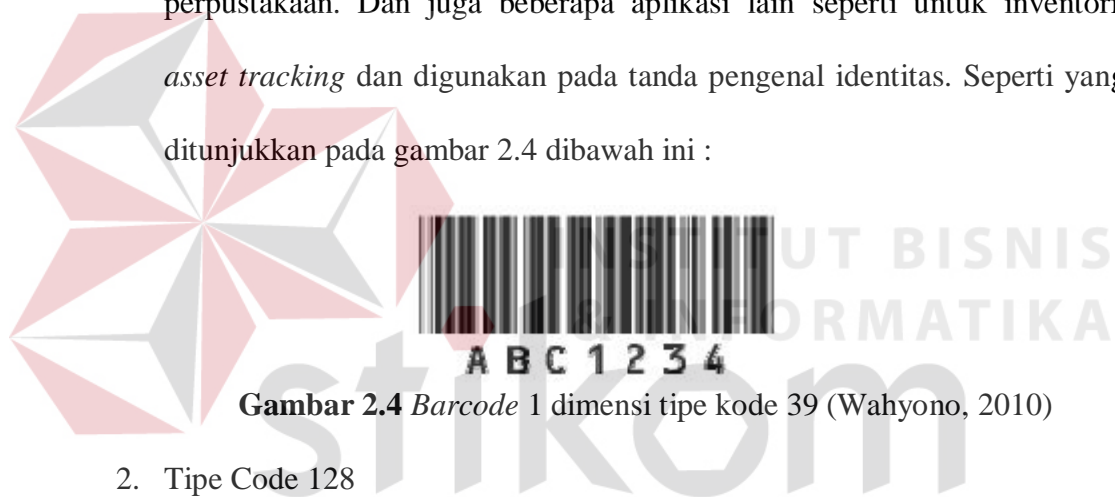
Peningkatan efisiensi dan produktivitas yang diberikan *barcode* memungkinkan perusahaan menghemat biaya dan waktu, sehingga profit perusahaan meningkat.

Menurut Wahyono (2010) *barcode* memiliki banyak sekali standard dan tidak terpaku pada satu model tertentu. Tetapi meski demikian, secara umum *barcode* terbagi menjadi dua kelompok besar, yaitu *barcode* satu dimensi dan *barcode* dua dimensi.

Barcode ini dinamakan satu dimensi atau ada juga yang menyebut *linear barcodes* karena kodenya hanya terdiri dari barisbaris. Berikut ini adalah beberapa jenis *barcode* satu dimensi :

1. Tipe code 39

Merupakan *barcode alphanumeric* (full ASCII) yang dapat mewakili abjad (A-Z) dan angka (0-9), serta beberapa karakter lain, seperti \$, /, +, % titik dan spasi dan juga memiliki jumlah digit maximal 16. Kode seperti ini biasa nya di gunakan untuk *barcode* buku maupun untuk *barcode* anggota perpustakaan. Dan juga beberapa aplikasi lain seperti untuk inventori, *asset tracking* dan digunakan pada tanda pengenalan identitas. Seperti yang ditunjukkan pada gambar 2.4 dibawah ini :



Gambar 2.4 *Barcode* 1 dimensi tipe kode 39 (Wahyono, 2010)

2. Tipe Code 128

Barcode tipe code 128 juga tergolong di dalam *barcode alphanumeric* (full ASCII), tetapi memiliki kerapatan yang lebih tinggi dan juga panjang baris yang bervariasi. *Barcode* Tipe Code 128 biasa nya di gunakan untuk aplikasi, seperti pengaturan maskapai pelayaran dan pengelolaan gudang. Setiap karakter pada code 128 47 dikodekan oleh 3 bar dan 3 spasi (atau 6 elemen) dengan ketebalan masing-masing elemen 1 sampai 4 kali ketebalan minimum (module). Seperti yang ditunjukkan pada gambar 2.5 dibawah ini :



Gambar 2.5 Barcode 1 dimensi tipe kode 128 (Wahyono, 2010)

3. Code 25 (*Interleaved*)

Merupakan kode *barcode* yang hanya untuk angka (0-9), maksimum 32 digit. Jadi *barcode* ini berbentuk numeric dan memiliki panjang baris yang bervariasi. *Barcode* yang juga disebut sebagai *interleaved 2 of 5* biasa digunakan untuk aplikasi dalam dunia industry dan laboratorium. . Seperti yang ditunjukkan pada gambar 2.6 dibawah ini :



Gambar 2.6 Barcode 1 dimensi tipe kode 25 (Wahyono, 2010)

4. EAN 13

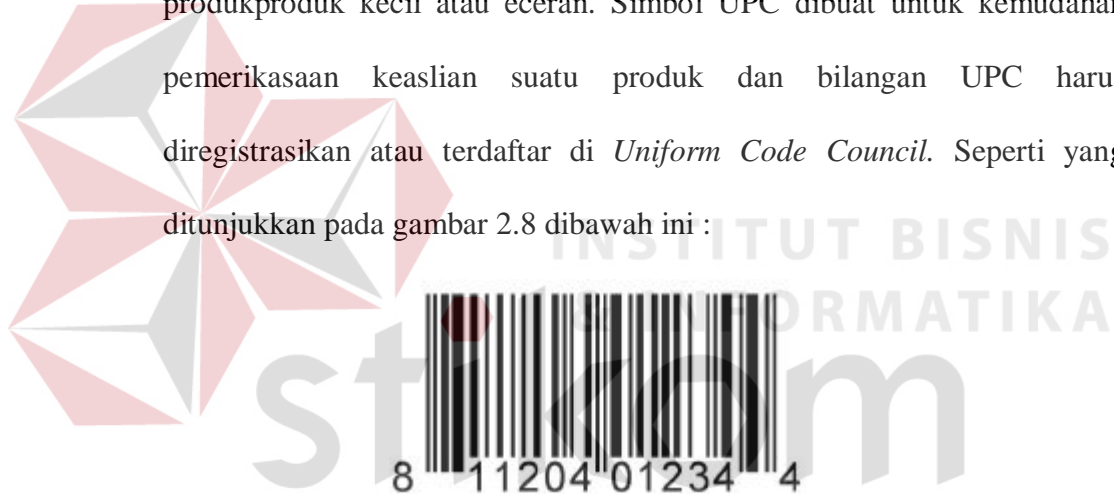
Simbologi *barcode* model ini dikeluarkan EAN untuk identitas suatu produk. Standarisasi EAN menggunakan 3 digit pertama adalah untuk kode Negara asal produk, 4 digit berikutnya adalah *Manufacture Number*, 5 digit berikutnya adalah kode produk atau nomor urut produk sedangkan 1 digit terakhir adalah cek digit atau angka untuk melakukan tes validasi *barcode*. Kode EAN sering digunakan di Indonesia untuk identifikasi produk nasional. Seperti yang ditunjukkan pada gambar 2.7:



Gambar 2.7 Barcode 1 dimensi tipe kode EAN 13 (Wahyono, 2010)

5. UPC (*Universal Product Code*)

Barcode UPC terdiri dari angka (0-9) namun *barcode* harus mempunyai panjang tepat 11 atau 12 digit, Kurang atau lebih dari angka tersebut tidak bisa digunakan. Jadi *barcode* ini berbentuk numerik dan memiliki panjang baris yang tetap. UPC biasanya digunakan untuk pelabelan pada produk-produk kecil atau eceran. Simbol UPC dibuat untuk kemudahan pemeriksaan keaslian suatu produk dan bilangan UPC harus diregistrasikan atau terdaftar di *Uniform Code Council*. Seperti yang ditunjukkan pada gambar 2.8 dibawah ini :



Gambar 2.8 Barcode 1 dimensi tipe kode UPC (Wahyono, 2010)

Barcode dua dimensi adalah *barcode* yang tidak hanya terbuat dari garis-garis saja, tetapi lebih mendekati bentuk gambar tertentu. Dengan menggunakan dua dimensi, informasi atau data yang besar dapat disimpan di dalam suatu ruang yang lebih kecil. Berikut ini adalah beberapa jenis *barcode* dua dimensi :

1. QR code

QR merupakan kependekan dari *Quick Response*, sebuah harapan dari pembuatnya bahwa code ini akan cepat di-decode. Tidak seperti *barcode* yang satu sisinya saja yang mengandung data, QR code mempunyai dua

sisi yang berisi data. Hal tersebut membuat QR code dapat lebih banyak memuat informasi dibandingkan *barcode*. Selain itu QR code juga dapat menampung informasi berupa URL suatu *website* yang nantinya dapat digunakan pada majalah, iklan, atau media lainnya, sehingga ketika seseorang memiliki handphone berkamera dan memiliki fitur pembaca, QR code dapat langsung men-scan dan masuk kedalam *website* yang dimaksud tanpa perlu mengetikkan alamatnya. Seperti yang ditunjukkan pada gambar 2.9 dibawah ini :



Gambar 2.9 Barcode 2 dimensi tipe QR (Wahyono, 2010)

2. Kode PDF417

Model ini merupakan pengembangan dari *barcode* satu dimensi linear, tepatnya disebut sebagai *stacked linear barcode*. PDF sendiri di artikan sebagai *Portable Data File*. Model ini memiliki kemampuan untuk menyimpan lebih dari 2000 karakter di dalam ruang berukuran 4 inch persegi. Seperti yang ditunjukkan pada gambar 2.10 dibawah ini :

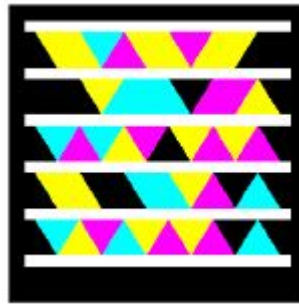


Gambar 2.10 Barcode 2 dimensi tipe PDF417 (Wahyono, 2010)

3. HCCB (*High Capacity Color Barcode*)

Contoh lain dari *barcode* dua dimensi adalah HCCB yang mana didevelop oleh Microsoft. Microsoft mengimplementasikan HCCB menggunakan 4

warna pada ukuran 5x10 *grid*. Dengan *barcode* HCCB data akan mampu disimpan dalam ukuran yang lebih besar. Selain itu tampilan kode juga lebih enak untuk dilihat karena menggunakan tampilan berwarna(*color barcode*). Seperti yang ditunjukkan pada gambar 2.11 dibawah ini :



Gambar 2.11 Barcode 2 dimensi tipe HCCB (Wahyono, 2010)

4. Data Matrix

Data Matrix adalah *barcode* dua dimensi yang terdiri dari sel hitam dan putih atau modul titik-titik di sekitar *rectangular patern*. *Data matrix* menggunakan ukuran data sampai pada 2 *kilobytes*. Sebuah *data matrix* dapat menyimpan sampai dengan 2.335 karakter *alphanumeric*. Salah satu contoh penggunaan *data matrix* ini yang sering dijumpai seperti *barcode* pada *Mini PCI Card* dengan ukuran symbol yang bervariasi mulai dari 8x8 sampai 144x144 seperti yang ditunjukkan pada gambar 2.12 dibawah ini :



Gambar 2.12 Barcode 2 dimensi tipe Data Matrix (Wahyono, 2010)