BAB 2

TINJAUAN PUSTAKA

2.1. Objek Penelitian

Kemajuan teknologi telah membawa perubahan-perubahan yang cepat dan signifikan pada industri farmasi, obat asli Indonesia, makanan, kosmetika dan alat kesehatan. Dengan menggunakan teknologi modern, industri-industri tersebut kini mampu memproduksi dalam skala yang sangat besar mencakup berbagai produk dengan *"range"* yang sangat luas.

Dengan dukungan kemajuan teknologi transportasi dan entry barrier yang makin tipis dalam perdagangan internasional, maka produk-produk tersebut dalam waktu yang sangat singkat dapat menyebar ke berbagai negara dengan jaringan distribusi yang sangat luas dan mampu menjangkau seluruh strata masyarakat.

Konsumsi masyarakat terhadap produk-produk termaksud cenderung terus meningkat, seiring dengan perubahan gaya hidup masyarakat termasuk pola konsumsinya. Sementara itu pengetahuan masyarakat masih belum memadai untuk dapat memilih dan menggunakan produk secara tepat, benar dan aman. Di lain pihak iklan dan promosi secara gencar mendorong konsumen untuk mengkonsumsi secara berlebihan dan seringkali tidak rasional.

Perubahan teknologi produksi, sistem perdagangan internasional dan gaya hidup konsumen tersebut pada realitasnya meningkatkan resiko dengan implikasi yang luas pada kesehatan dan keselamatan konsumen. Apabila terjadi produk sub standar, rusak atau terkontaminasi oleh bahan berbahaya maka risiko yang terjadi akan berskala besar dan luas serta berlangsung secara amat cepat.

Untuk itu Indonesia harus memiliki Sistem Pengawasan Obat dan Makanan (SisPOM) yang efektif dan efisien yang mampu mendeteksi, mencegah dan mengawasi produk-produk termaksud untuk melindungi keamanan, keselamatan dan kesehatan konsumennya baik di dalam maupun di luar negeri. Untuk itu telah

dibentuk Badan POM yang memiliki jaringan nasional dan internasional serta kewenangan penegakan hukum dan memiliki kredibilitas profesional yang tinggi.

2.1.1. Visi Dan Misi

Visi

Menjadi Institusi Pengawas Obat dan Makanan yang Inovatif, Kredibel dan Diakui Secara Internasional Untuk Melindungi Masyarakat.

Misi

1. Melakukan Pengawasan *Pre-Market* dan *Post-Market* Berstandar Internasional.
2. Menerapkan Sistem Manajemen Mutu Secara Konsisten.
3. Mengoptimalkan Kemitraan dengan Pemangku Kepentingan di Berbagai Lini.
4. Memberdayakan Masyarakat Agar Mampu Melindungi Diri dari Obat dan Makanan yang Berisiko Terhadap Kesehatan.
5. Membangun Organisasi Pembelajar *(Learning Organization).*

2.1.2. Fungsi

1. Pengkajian dan penyusunan kebijakan nasional di bidang pengawasan Obat dan Makanan.
2. Pelaksanaan kebijakan tertentu di bidang pengawasan Obat dan Makanan.
3. Koordinasi kegiatan fungsional dalam pelaksanaan tugas Badan POM.
4. Pemantauan, pemberian bimbingan dan pembinaan terhadap kegiatan instansi pemerintah di bidang pengawasan Obat dan Makanan.
5. Penyelenggaraan pembinaan dan pelayanan administrasi umum di bindang perencanaan umum, ketatausahaan, organisasi dan tata laksana, kepegawaian, keuangan, kearsipan, persandian, perlengkapan dan rumah tangga.

2.1.3. Budaya Organisasi

Budaya organisasi merupakan nilai-nilai luhur yang diyakini dan harus dihayati dan diamalkan oleh seluruh anggota organisasi dalam melaksanakan tugas. Nilai-nilai luhur yang hidup dan tumbuh kembang dalam organisasi menjadi semangat bagi seluruh anggota organisasi dalam berkarsa dan berkarya.

1. Profesional

   Menegakkan profesionalisme dengan integritas, objektivitas, ketekunan dan komitmen yang tinggi.

2. Kredibel

   Dapat dipercaya dan diakui oleh masyarakat luas, nasional dan internasional.

3. Cepat Tanggap

   Antisipatif dan responsif dalam mengatasi masalah.

4. Kerjasama Tim

   Mengutamakan keterbukaan, saling percaya dan komunikasi yang baik.

5. Inovatif

   Mampu melakukan pembaruan sesuai ilmu pengetahuan dan teknologi terkini.

2.1.4. Target Kinerja

1. Terkendalinya penyaluran produk terapetik dan NAPZA
2. Terkendalinya mutu, keamanan dan khasiat/kemanfaatan produk obat dan makanan termasuk klim pada label dan iklan di peredaran;
3. Tercegahnya risiko penggunaan bahan kimia berbahaya sebagai akibat pengelolaan yang tidak memenuhi syarat;
4. Penurunan kasus pencemaran pangan;
5. Peningkatan kapasitas organisasi yang didukung dengan kompetensi dan keterampilan personil yang memadai;
6. Terwujudnya komunikasi yang efektif dan saling menghargai antar sesama dan pihak terkait.

2.1.5. Prinsip Dasar SisPOM

1. Tindakan pengamanan cepat, tepat, akurat dan profesional.
2. Tindakan dilakukan berdasarkan atas tingkat risiko dan berbasis bukti-bukti ilmiah.
3. Lingkup pengawasan bersifat menyeluruh, mencakup seluruh siklus proses.
4. Berskala nasional/lintas propinsi, dengan jaringan kerja internasional.
5. Otoritas yang menunjang penegakan supremasi hukum.
6. Memiliki jaringan laboratorium nasional yang kohesif dan kuat yang berkolaborasi dengan jaringan global.
7. Memiliki jaringan sistem informasi keamanan dan mutu produk.

2.1.6. Kerangka Konsep SisPOM

Pengawasan obat dan makanan memiliki aspek permasalahan berdimensi luas dan kompleks. Oleh karena itu diperlukan sistem pengawasan yang komprehensip, semenjak awal proses suatu produk hingga produk tersebut beredar ditengah masyarakat. Untuk menekan sekecil mungkin risiko yang bisa terjadi, dilakukan SISPOM tiga lapis yakni:

1. Sub-sistem pengawasan Produsen

   Sistem pengawasan internal oleh produsen melalui pelaksanaan cara-cara produksi yang baik atau *good manufacturing practices* agar setiap bentuk penyimpangan dari standar mutu dapat dideteksi sejak awal. Secara hukum produsen bertanggung jawab atas mutu dan keamanan produk yang dihasilkannya. Apabila terjadi penyimpangan dan pelanggaran terhadap standar yang telah ditetapkan maka produsen dikenakan sangsi, baik administratif maupun pro-justisia.

2. Sub-sistem pengawasan Konsumen

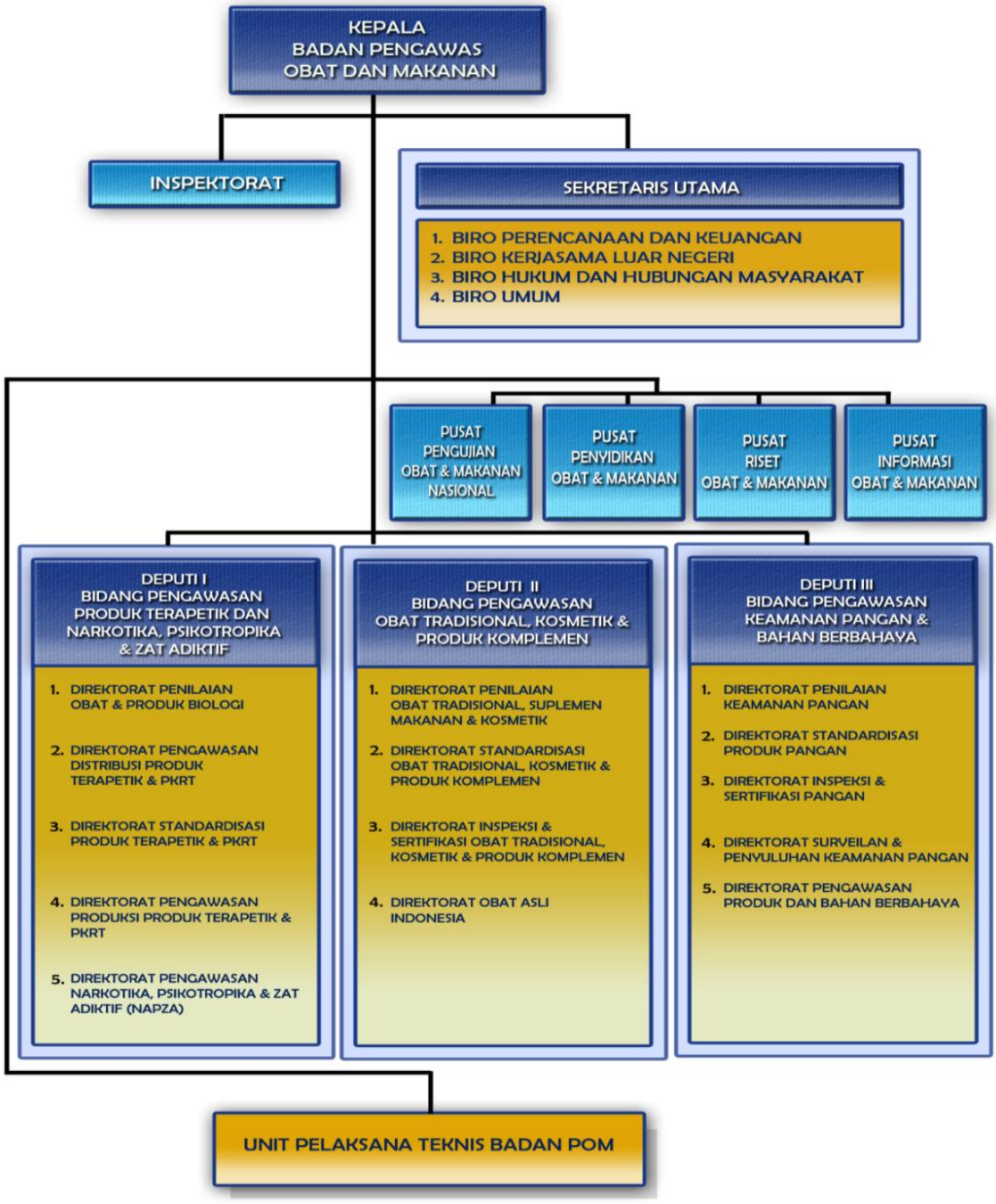
   Sistem pengawasan oleh masyarakat konsumen sendiri melalui peningkatan kesadaran dan peningkatan pengetahuan mengenai kualitas produk yang digunakannya dan cara-cara penggunaan produk yang rasional. Pengawasan

oleh masyarakat sendiri sangat penting dilakukan karena pada akhirnya masyarakatlah yang mengambil keputusan untuk membeli dan menggunakan suatu produk. Konsumen dengan kesadaran dan tingkat pengetahuan yang tinggi terhadap mutu dan kegunaan suatu produk, di satu sisi dapat membentengi dirinya sendiri terhadap penggunaan produk-produk yang tidak memenuhi syarat dan tidak dibutuhkan sedang pada sisi lain akan mendorong produsen untuk ekstra hati-hati dalam menjaga kualitasnya.

3. Sub-sistem pengawasan Pemerintah/Badan POM

Sistem pengawasan oleh pemerintah melalui pengaturan dan standardisasi; penilaian keamanan, khasiat dan mutu produk sebelum diijinkan beredar di Indonesia; inspeksi, pengambilan sampel dan pengujian laboratorium produk yang beredar serta peringatan kepada publik yang didukung penegakan hukum. Untuk meningkatkan kesadaran dan pengetahuan masyarakat konsumen terhadap mutu, khasiat dan keamanan produk maka pemerintah juga melaksanakan kegiatan komunikasi, informasi dan edukasi.

2.1.7. Struktur Organisasi



Gambar 2.1 Struktur Organisasi

2.2. String Matching

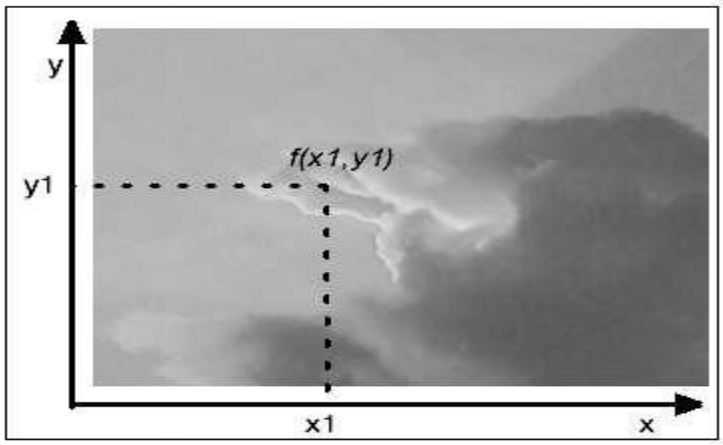
Pengertian string menurut *Dictionary of Algorithms and Data Structures, National Institute of Standards and Technology* (NIST) adalah susunan dari karakter-karakter (angka,alfabet atau karakter yang lain) dan biasanya direpresentasikan sebagai struktur dan array. String dapat berupa kata, frase, atau kalimat. Pencocokan string (string matching) merupakan bagian penting dari sebuah proses pencarian string(string searching) dalam sebuah dokumen. Hasil dari sebuah sebuah pencarian string dalam dokumen tergantung dari teknik dan cara pencocokan string yang digunakan.

String matching secara umum adalah pencarian sebuah *pattern* pada sebuah teks. *string matching* memiliki prinsip kerja adalah sebagai berikut:

1. Men-*scan teks* dengan bantuan sebuah *window* yang ukurannya sama dengan panjang *pattern.*
2. Menempatkan *window* pada awal *teks.*
3. Membandingkan karakter pada *window* dengan karakter dari *pattern.* Setelah pencocokan (baik hasilnya cocok atau tidak cocok), dilakukan *shft* ke kanan pada *window.* Prosedur ini dilakukan berulang-ulang sampai *window* berada pada akhir teks. Mekanisme ini disebut mekanisme *sliding-window.*

2.3. Citra Digital

Citra digital dapat didefinisikan sebagai fungsi dua variabel, f(x,y), dimana x dan y adalah koordinat spasial dan nilai f(x,y) adalah intensitas citra pada koordinat tersebut, hal tersebut diilustrasikan pada gambar 2.2.

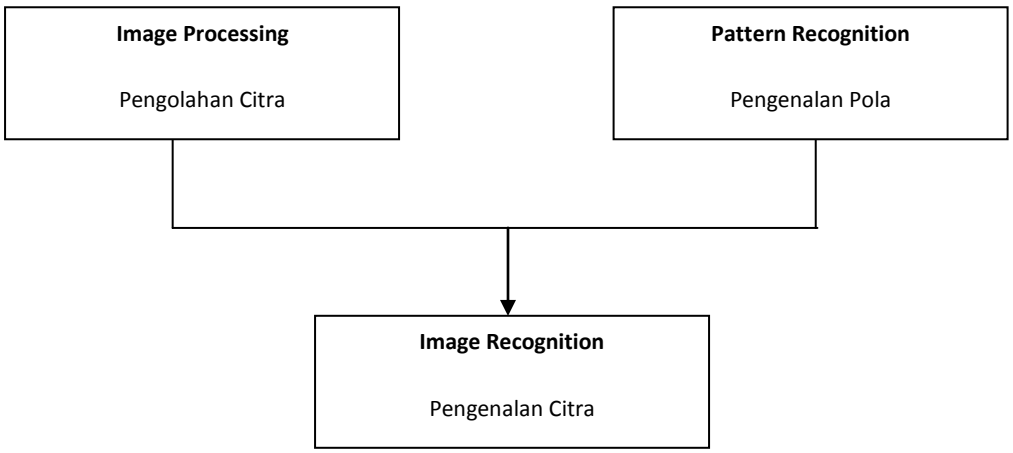


Gambar 2.2 Citra Digital

2.4. Pengolahan Citra

Secara harafiah, citra (*image*) adalah gambar pada bidang dwimatra (dua dimensi). Ditinjau dari sudut pandang matematis, citra merupakan fungsi menerus (*continue*) dari intensitas cahaya pada bidang dwimatra. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh oleh alat-alat optik, misalnya mata pada manusia, kamera, pemindai (*scanner*), dan sebagainya, sehingga bayangan objek yang disebut citra tersebut terekam [3].

Bila pengenalan pola dihubungkan dengan pengolahan citra, diharapkan akan terbentuk suatu sistem yang dapat memproses citra masukan sehingga citra tersebut dapat dikenali polanya. Proses ini disebut pengenalan citra atau *image recognition.* Proses pengenalan citra ini sering diterapkan dalam kehidupan sehari-hari.



Gambar 2.3 Bagan Pengolahan Citra

Pengolahan citra dan pengenalan pola menjadi bagian dari proses pengenalan citra. Kedua aplikasi ini akan saling melengkapi untuk mendapatkan ciri khas dari suatu citra yang hendak dikenali [4]. Secara umum tahapan pengolahan citra digital meliputi akusisi citra, peningkatan kualitas citra, segmentasi citra, representasi dan uraian, pengenalan dan interpretasi.

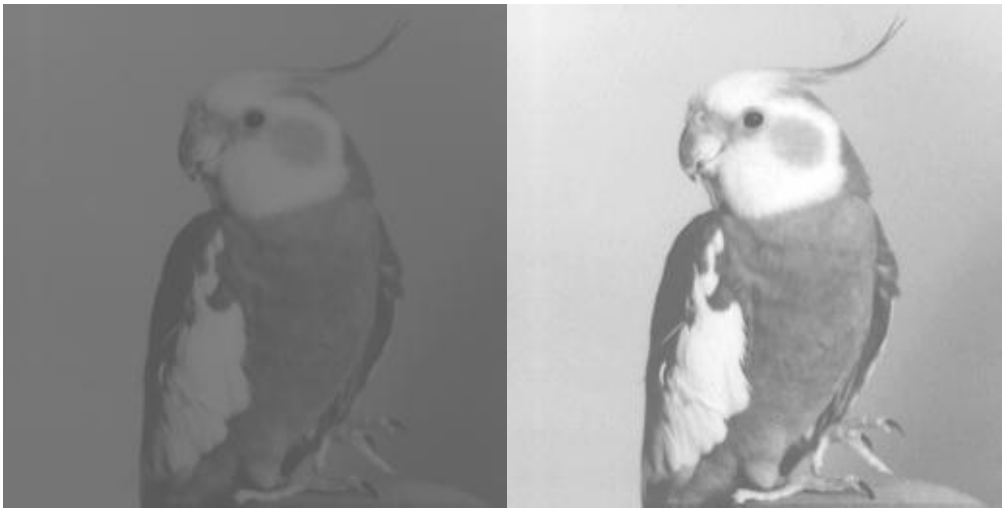
2.3.1. Pengertian Pengolahan Citra

Meskipun sebuah citra kaya informasi, namun seringkali citra yang kita miliki mengalami penurunan mutu (degradasi), misalnya mengandung cacat atau derau (*noise*), warnanya terlalu kontras, kurang tajam, kabur (*blurring*), dan sebagainya. Tentu saja citra semacam ini menjadi lebih sulit diinterpretasi karena informasi yang disampaikan oleh citra tersebut menjadi berkurang.

Agar citra yang mengalami gangguan mudah diinterpretasi (baik oleh manusia maupun mesin), maka citra tersebut perlu dimanipulasi menjadi citra lain yang kualitasnya lebih baik. Bidang studi yang menyangkut hal ini adalah pengolahan citra (*image processing*).

Pengolahan citra adalah pemrosesan citra, khususnya dengan menggunakan komputer, menjadi citra yang kualitasnya lebih baik. Sebagai contoh, citra burung

nuri pada Gambar 2.5(a) tampak agak gelap, lalu dengan operasi pengolahan citra kontrasnya diperbaiki sehingga menjadi lebih terang dan tajam (b).



(a) **(b)**

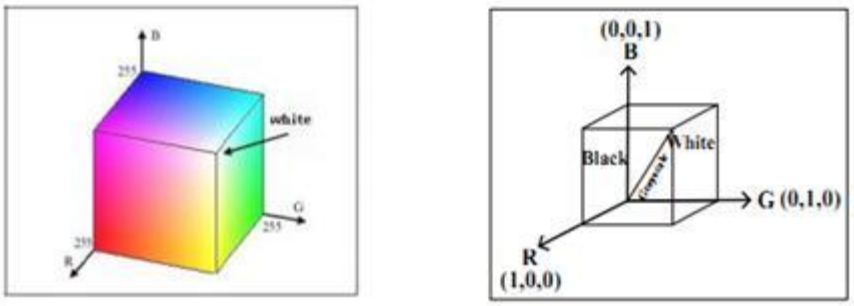
Gambar 2.4 (a) Citra burung nuri yang agak gelap, (b) Citra burung yang telah diperbaiki kontrasnya.

2.3.2. Penganalan Warna Citra Warna RGB

Warna adalah spektrum tertentu yang terdapat di dalam suatu cahaya sempurna (berwarna putih). Nilai warna ditentukan oleh tingkat kecerahan maupun kesuraman warna. Nilai ini dipengaruhi oleh penambahan putih ataupun hitam [5]

Penelitian memperlihatkan bahwa kombinasi warna yang memberikan rentang paling lebar adalah red (R), green (G) dan blue (B). Ketiga warna tersebut merupakan warna pokok yang biasa disebut RGB. Warna lain dapat diperoleh dengan mencampurkan ketiga warna pokok tersebut dengan perbandingan tertentu. Setiap warna pokok mempunyai intensitas sendiri dengan nilai maksimum 255 (8-bit). Misal warna kuning merupakan kombinasi warna merah dan hijau sehingga nilai RGB: 255 255 0.

RGB disebut juga ruang warna yang dapat divisualisasikan sebagai sebuah kubus seperti gambar 2.6, dengan tiga sumbunya yang mewakili komponen warna merah (red) R, hijau (green) G, biru (blue) B. Salah satu pojok alasnya yang Sistem Klasifikasi Jenis dan Kematangan Buah Tomat Berdasarkan Bentuk dan Ukuran serta Warna Permukaan Kulit Buah Berbasis Pengolahan Citra Digital berlawanan menyatakan warna hitam ketika R = G = B = 0, sedangkan pojok atasnya yang berlawanan menyatakan warna putih ketika R= G= B= 255 (sistem warna 8 bit bagi setiap komponennya).



Gambar 2.5 Ruang Warna RGB

Kamera yang digunakan pada Handphone ini akan menangkap gambar dari objek dan kemudian merepresentasikannya kedalam bentuk citra warna RGB untuk kemudian dilakukan pengolahan citra tersebut sehingga menghasilkan data-data objek yang dibutuhkan.

2.4. Jenis-jenis Citra Digital

Ada banyak cara untuk menyimpan citra digital di dalam memori. Cara penyimpanan menentukan jenis citra digital yang terbentu. Beberapa jenis citra digital yang sering digunakan adalah citra biner, citra grayclale, dan citra warna.

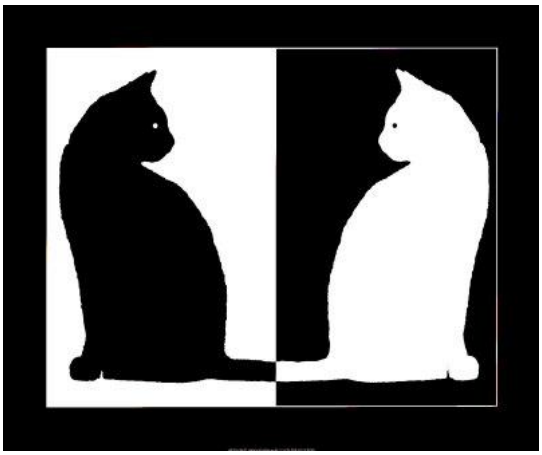
2.4.1. Citra Biner (Monokrom)

Citra biner (*binary image*) adalah citra yang hanya mempunyai dua nilai derajat keabuan: hitam dan putih seperti yang ditunjukkan pada gambar 2.7. Meskipun saat

ini citra berwarna lebih disukai karena memberi kesan yang lebih kaya daripada citra biner, namun tidak membuat citra biner mati. Pada beberapa aplikasi citra biner masih tetap dibutuhkan, misalnya citra logo instansi (yang hanya terdiri atas warna hitam dan putih), citra kode batang (*bar code*) yang tertera pada label barang, citra hasil pemindaian dokumen teks, dan sebagainya [5]

Proses awal yang dilakukan dalam menganalisis objek di dalam citra biner adalah segmentasi objek. Proses segmentasi bertujuan mengelompokkan *pixel-pixel* objek menjadi wilayah (*region*) yang merepresentasikan objek. Ada dua pendekatan yang digunakan dalam segmentasi objek:

1. Segmentasi berdasarkan batas wilayah (tepi dari objek). *Pixel-pixel* tepi ditelusuri sehingga rangkaian *pixel* yang menjadi batas (*boundary*) antara objek dengan latar belakang dapat diketahui secara keseluruhan (algoritma *boundary following*).
2. Segmentasi ke bentuk-bentuk dasar (misalnya segmentasi huruf menjadi garis-garis vertikal dan horizontal, segmentasi objek menjadi bentuk lingkaran, elips, dan sebagainya).



Gambar 2.6. Citra Biner

Pada penelitian ini, citra biner digunakan untuk menyimpan hasil pendeteksian objek berdasarkankan warna objeknya. Pemilihan menggunakan citra biner sebagai tempat penyimpanan data sementara untuk mempercepat proses pendeteksian pola

objek yang dilakukan berikutnya dikarenakan ukuran memori yang digunakan pada citra biner sangat kecil, yaitu 1 bit (*true* dan *false*).

2.4.2. Citra Warna (*True Color*)

Pada citra warna setiap titik mempunyai warna yang spesifik, yang merupakan kombinasi dari 3 warna dasar, yaitu merah, hijau, dan biru. Format citra ini sering disebut RGB (*red-green-blue*). Setiap warna dasar mempunyai intensitas sendiri dengan nilai maksimum 255 (8 bit). Setiap titik pada citra warna membutuhkan data 3 *byte*.

Jumlah kombinasi warna yang memungkinkan untuk format citra ini adalah 2^{24}atau lebih dari 16 juta warna, dengan demikian bisa dianggap mencakup semua warna yang ada. Inilah sebabnya format ini dinamakan *true color*.

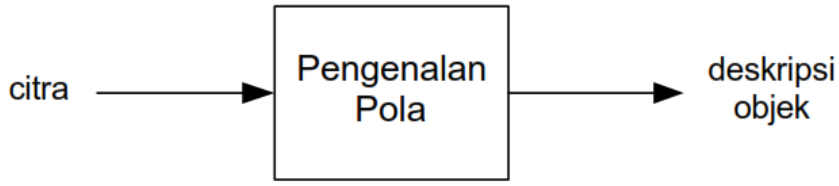


= 255 255 255 255 0 0 128 128 128 128 128 0
= 0 255 255 0 128 0 204 255 255 0 0 255
= 150 150 150 51 51 51 255 255 255 95 95 95
= 255 204 153 204 255 204 128 0 0 255 0 255

Gambar 2.7 Citra warna (true color)

2.4.3. Pengenalan Pola

Pengenalan pola adalah mengelompokkan data numerik dan simbolik (termasuk citra) secara otomatis oleh mesin (dalam hal ini komputer). Tujuan pengelompokan adalah untuk mengenali suatu objek di dalam citra. Manusia bisa mengenali objek yang dilihatnya karena otak manusia telah belajar mengklasifikasi objek-objek di alam sehingga mampu membedakan suatu objek dengan objek lainnya. Kemampuan sistem visual manusia inilah yang dicoba ditiru oleh mesin. Komputer menerima masukan berupa citra objek yang akan diidentifikasi, memproses citra tersebut, dan memberikan keluaran berupa deskripsi objek di dalam citra [4].



Pola adalah entitas yang terdefinisi dan dapat didefinisikan melalui ciri-cirinya (feature). Ciri-ciri tersebut digunakan untuk membedakan suatu pola dengan pola yang lain. Pengenalan pola bertujuan untuk menentukan kelompok untuk kategori pola berdasarkan ciri-ciri yang dimiliki oleh pola tersebut. Dengan kata lain pengenalan pola membedakan objek dengan objek lain.

Suatu sistem pengenalan pola melakukan:

1. Proses akuisisi data melalui sejumlah alat pengindraan atau sensor,
2. Mengatur bentuk representasi data,
3. Melakukan proses analisis dan klasifikasi data.

Tiga pendekatan pembuatan sistem pengenalan pola adalah Statistik (statistical), Sintaksis (syntactic) dan Jaringan Saraf Tiruan (neural network) (Schalkoff, 1992).

1. Statistik

Semakin banyak pola yang disimpan, maka sistem akan semakin cerdas. Salah satu contoh penerapannya banyak pada pola pengenalan iris scan.

Kelemahannya: hanya bergantung pada data yang disimpan saja, tidak memiliki sesuatu struktur yang unik yang dapat menjadi kunci pengenalan pola.

2. Sintaksis (rule)

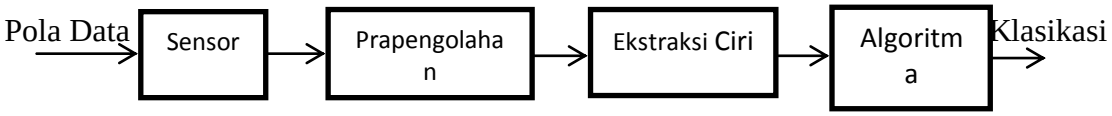
Dengan rule/aturan maka sistem yang lebih terstruktur sehingga memiliki sesuatu ciri yang unik. Salah satu contoh penerapannya pada pola pengenalan sidik jari (fingerprint).

3. Jaringan Saraf Tiruan (JST)

Merupakan gabungan dari pendekatan statistik dan pendekatan sintaks. Dengan gabungan dari dua metode maka JST merupakan pengenalan pola yang lebih akurat. Salah satu contoh penerapannya pada pola pengenalan suatu citra JST merupakan

suatu sistem yang dapat memproses informasi dengan meniru cara kerja jaringan saraf otak manusia.

Struktur sistem pengenalan pola ditunjukkan pada gambar 2.8. Sistem pengenalan pola ini terdiri dari suatu sensor (misalnya kamera, dan scanner), teknik prapengolahan, suatu algoritma atau mekanisme ekstraksi ciri dan algoritma untuk klasifikasi atau pengenalan (bergantung pada pendekatan yang dilakukan). Sebagai tambahan, biasanya beberapa data yang sudah diklasifikasikan diasumsikan telah tersedia untuk melatih sistem.

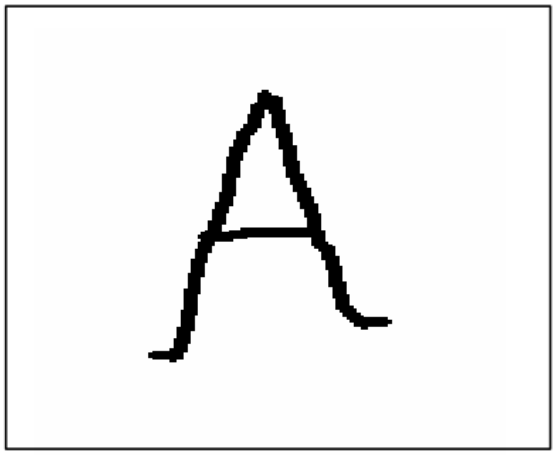


Gambar 2.8. Struktur Sistem Pengenalan Pola

Prapengolahan adalah transformasi input (masukan) data mentah untuk membantu kemampuan komputasional dan pencarian ciri serta untuk mengurangi noise (derau). Pada prapengolahan citra (sinyal) yang ditangkap oleh sensor akan dinormalisasi agar citra menjadi lebih siap untuk diolah pada tahap pemisahan ciri. Kualitas ciri yang dihasilkan pada proses pemisahan ciri sangat bergantung pada hasil prapengolahan.

Klasifikasi merupakan tahap untuk mengelompokkan input data pada satu atau beberapa kelas berdasarkan hasil pencarian beberapa ciri yang signifikan dan pemrosesan atau analisis terhadap ciri itu. Setiap kelas terdiri dari sekumpulan objek yang memiliki kedekatan (kemiripan) ciri. (Munir, 2004; Putra, 2009).

Contoh pengenalan pola misalnya citra pada Gambar 2.9 adalah tulisan tangan yang digunakan sebagai data masukan untuk mengenali karakter '*A*'. Dengan menggunakan suatu algoritma pengenalan pola, diharapkan komputer dapat mengenali bahwa karakter tersebut adalah '*A*'.



Gambar 2.9 Pola 2 Dimensi

2.5. Elemen- Elemen Citra Digital

Citra digital mengandung sejumlah elemen-elemen dasar.Elemen-elemen dasar tersebut dimanipulasi dalam pengolaha citra dan dieksploitasi lebih lanjut dalam komputer vision . Elemen elemen dasar diantaranya:

2.5.1. Kecerahan (brightness)

Kecerahan adalah kata lain untuk intensitas cahaya. Sebagai mana telah dijelaskan pada bagian sampling, kecerahan pada sebuah titik (pixel) didalam citra bukanlah intensitas yang riil, tetapi sebenarnya adalah intensitas rata-rata dari suatu area yang melingkupinya. Sistem visual manusia mampu menyesuaikan dirinya dengan tingkatan kecerahan (brightness level) mulai dari yang paling rendah sampai yang paling tinggi dengan jangkauan 1010.

2.5.2. Kontras (contrast)

Kontras menyatakan sebaran terang (lightness) dan gelap (darkness) didalam sebuah gambar. Citra dengan kontras rendah dicirikan sebagai besar komposisi citranya adalah terang sebagian besar gelap. Pada citra dengan kontras yang baik, komposisi gelap dan terang tersebar secara merata.

2.5.3. Kontur (contour)

Kontur adalah keadaan yang ditimbulkan oleh perubahan intensitas cahaya pada pexel pixel yang bertetangga. Karena adanya perubahan intensitas inilah mata kita mampu mendeteksi tepi-tepi (edge) objek didalam citra.

2.5.4. Warna (color)

Warna adalah persepsi yang dirasakan oleh sistem visual manusia terhadap panjang gelombang cahaya yang dipantulkan oleh objek. Setiap warna mempunyai panjang gelombang (λ) yang berbeda. Warna merah mempunyai panjang gelombang yang paling tinggi, sedangkan warna ungu (violet) mempunyai panjang gelombang yang paling rendah.Warna-warna yang diterima oleh mata (sistem visual mata) merupakan hasil kombinasi cahaya dengan panjang gelombang yang berbeda. Penelitian memperlihatkan kombinasi warna yang memberikan rentang warna yang paling lebar adalah merah (red), hijau (green), biru (blue). Penyesuaian warna pada visual kita tidak jarang dapat menimbulkan "cacat" warna (distorsi) yang dilihat. Ada dua jenis distorsi, yakni distorsi warna terhadap ruang (misal bercak abu-abu yang berada disekitar warna hijau akan berkesan ungu), dan distorsi terhadap waktu (misalnya setelah melihat warna hijau kita langsung melihat warna abu-abu, maka warna ungulah yang berkesan pada mata kita).

2.5.5. Bentuk (shape)

Pada umumnya citra yang dibentuk oleh mata merupakan citra dua dimensi, sedangkan objek yang diamati biasanya adalah 3 dimensi telah diproyeksikan kebidang dua dimensi dan kelihatannya sama. Misalnya, suatu ruangan terlihat berbentuk trapezium pada gambar dua dimensi. Didalam hal ini kita tahu apakah hal ini memang disebabkan oleh bentuk ruangan yang panjang ataukah memang ruangan tersebut berbentuk trapesium.

2.5.6. Tekstur (texture)

Pada hakikatnya sistem visual manusia tidak menerima informasi citra secara terpisah pada setiap titik, tetapi sesuatu citra dianggapnya sebagai suatu kesatuan, jadi

definisi kesamaan suatu objek perlu dinyatakan dalam bentuk kesamaan dari suatu himpunan parameter citra (brightness, color, size) atau dengan kata lain dua buah citra tidak dapat disamakan dari satu parameter saja.

2.6. Tresholding

Tresholding digunakan untuk membedakanmgambar text dengan latar belakang pada gambar huruf atau angka tersebut. Proses ini akan menghasilkan citra hitam putih yang bersih dari tingkat keabuan (grayscale), atau dengan kata lain metode ini mengkonversi citra gray-level ke citra bilevel (binary image). Untuk mendapatkan citra grayscale digunakan persamaan berikut:

$$I_{grayscale}(x,y) = \alpha I_{colour}(x,y,r) + \beta\, I_{colour}(x,y,g) + \gamma I_{colour}(x,y,b)$$

Dimana :

$I_{grayscale}$ = citra grayscale

I_{colour} = citra RGB

(x,y) = koordinat citra

(x,y,c) = channel piksel pada kordinat (x,y), r untuk merah, b untuk biru dan g untuk hijau α, β, γ = koefisien Pembobotan nilai koefisien (α, β dan γ) berdasarkan nilai dari respon mata manusia, biasanya ketiga nilai yang digunakan adalan 0.333 [20].

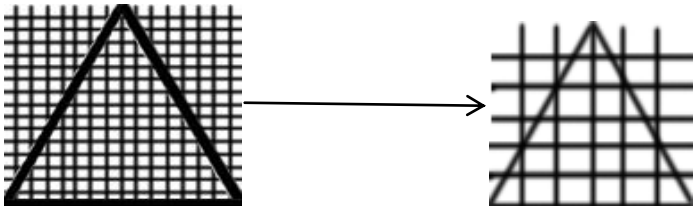
Setelah mendapatkan citra grayscale, citra biner dapat dibentuk dengan teknik thresholding. Jika g(x, y) adalah sebuah nilai ambang (threshold) batas dari f(x, y) yang terdapat pada gambar 2.2 dengan nilai threshold T. Nilai T digunakan untuk memisahkan antara object dengan backgroundnya, maka hasil threshold dapat ditulis sebagai berikut:

$$g(x,y) = \begin{cases} 1 & f(x,y) \geq T \\ 0, & f(x,y) < T \end{cases}$$

2.7. Pencocokan pola

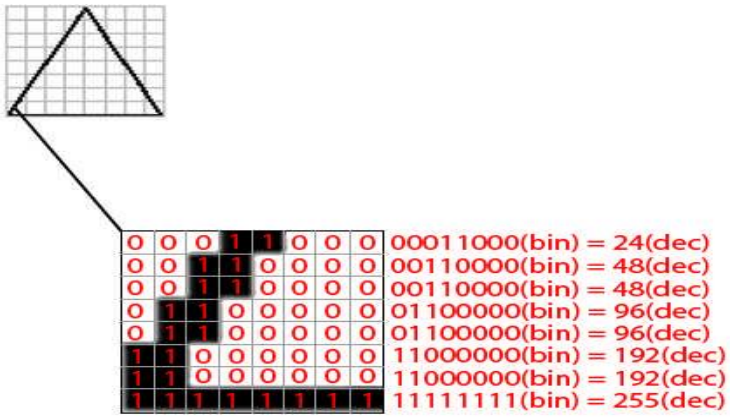
Setelah semua area persegi dan pola-pola gambar ditandai, menganalisa citra yang berada di dalam persegi dan membandingkan polanya dengan sekumpulan pola yang telah ditentukan (pencocokan pola). Dan mengekstrak pola didalam persegi menggunakan *transformasi homography*. Kemudian memberikan memberikan nilai *confidance* kepada pola yang cocok, jika kecocokannya di atas nilai yang telah ditentukan maka polanya dinyatakan cocok.

Pada proses ini sistem akan menggunakan metode scaling yang dimaksudkan untuk memperkecil (*zoom-out)* citra. Metode scaling digunakan untuk melakukan transformasi agar objek yang dimaksud dapat dimuat dalam wadah berukuran 64x64 pixel sebagai ukuran wadah untuk proses pengenal objek.



Gambar 2.10 Proses pencocokan zoom out

Hingga tahap ini citra objek yang akan dikenali sudah dalam bentuk citra biner dan berukuran 64x64 pixel, sehingga siap untuk dilakukan proses segementasi menjadi 64 buah segment berukuran 8x8 pixel.



Gambar 2.11 Proses deret biner desimal

Proses ini menjadi deret desimal dengan melakukan konversi setiap baris pada segment yang merupakan deret biner menjadi sebuah nilai desimal.

2.8. OCR

OCR (Optical Character Recognition) atau sering disebut dengan Pengenalan Pola karakter yang merupakan salah satu cabang dari pengenalan pola. Salah satu metode pengenalan pola angka atau huruf banyak digunakan algoritma Neural Network Backpropagation, yang sistem kerjanya seperti otak manusia yang dapat mengenali pola setelah mendapatkan pelatihan atau pengecekan terhadap beberapa contoh terlebih dahulu.

Teknologi OCR menggunakan dua metode, yaitu matric matching dan feature extraction. Dari dua metode ini, matric matching adalah teknik yang paling simple dalam mendeteksi karakter. Selain itu juga terbatas, sehingga keakuratannya kerap lebih rendah dibandingkan feature extraction.

Matric Matching dan Feature Extraction Metode pertama dan yang paling simple dalah Matrix Matching. Lebih dikenal sebagai pattern matching (String Matching). Ini dikarenakan scanner akan mendeteksi karakter yang dipindai lalu menyamakan dengan database dari pola atau outline dari setiap karakter huruf yang disimpannya dalam bentuk bitmap image. Jadi ketika scanner memindai sebuah huruf

maka aplikasi akan mendeteksi bentuk huruf tersebut dan menyamakan bentuk, sudut, dan pola karakter dari gambar bitmap yang ada di database. Jika ada yang cocok maka huruf yang dipindai tersebut akan diidentifikasi sebagai huruf yang sama dalam database. Hanya saja teknik ini memiliki keterbatasan, karena database ini memiliki keterbatasan akan jumlah variasi bentuk karakter. Karena, font saat ini beraneka ragam , dan bentuknya ada yang semakin kompleks dan beraneka pola.

Metode kedua, Feature Extraction lebih pintar. Dikenal juga sebagai ICR atau Intelligent Character Recognition atau topological feature analysis. Ini adalah metode pengenalan karakter yang tidak tergantung dari template karakter yang dimiliki aplikasi. Aplikasi dengan metode ini akan mendeteksi identitas sebuah karakter dari ruang kosong, bentuk yang berdekatan, garis diagonal, perpotongan dan sebagainya.Sehingga pengidentifikasiannya lebih kompleks prosesnya. Feature Extraction biasanya digunakan untuk mendeteksi karakter yang bentuknya tak standar, seperti tulisan tangan atau huruf-huruf berornamen. Matric Matching cocoknya untuk document yang menggunakan jenis huruf yang umum, seperti Times New Romans, Arial, atau font yang memiliki bentuk dasar karakter huruf lainnya. Database karakter ini biasanya meliputi rangkaian dasar karakter angka dari 0-9 dan huruf a-z.

Cara kerja OCR terdiri dari modul yang terpisah yang digunakan dalam proses pengenalan, modul-modul tersebut adalah:

1) Pra Pengolahan (Preprocessing)

   Merupakan suatu proses untuk menghilangkan bagian-bagian yang tidak diperlukan pada gambar input untuk proses selanjutnya. Contoh preprocessing adalah noise filtering.

2) Klasterisasi (Clusterization)

   Modul klasterisasi bertugas untuk menemukan karakter individu dalam gambar dan menghitung urutan karakter yang benar dalam teks.

3) Ekstraksi Fitur (Feature extraction)

   Tujuan dari modul ini adalah untuk mengekstrak fitur unik dari karakter individu sehingga dapat diakui oleh modul klasifikasi. Jika kita membandingkan sistem OCR dengan manusia, kita dapat mengatakan bahwa modul ini memainkan peran mata manusia.

4) Klasifikasi (Classification)

   Pada tahap terakhir sistem OCR mencoba untuk mengenali karakter menggunakan informasi tentang fitur karakter yang telah diekstraksi pada tahap sebelumnya. Meskipun dapat dilakukan secara statistik dengan melibatkan perbandingan konsisten karakter dari database, hal ini dipilih untuk mencoba bagaimana jaringan saraf melakukan pengenalan pada tahap ini.

2.9. Neural Network Backpropagation

Neural Network (NN) adalah suatu model komputasi dengan karakteristik yang menyerupai jaringan saraf manusia. NN terdiri dari unit-unit terkecil yang meproses sinyal input yaitu neuron. Pemrosesan data berupa sinyal dilakukan pada neuron tersebut.

Sinyal pada NN dikirim antara neuron melalui penghubung antar neuron memiliki bobot, sinyal yang melalui penghubung antara neuron akan dikalikan dengna bobotnya. Lalu setelah sinyal dikalikan dengan bobotnya, sinyal tersebut dijumlahkan dengan sinyal-sinyal yang lain yang juga telah dikalikan dengan bobot penghubung antar neuron. Kemudian menjadi nilai input bagi neuron pada layer yang dituju.

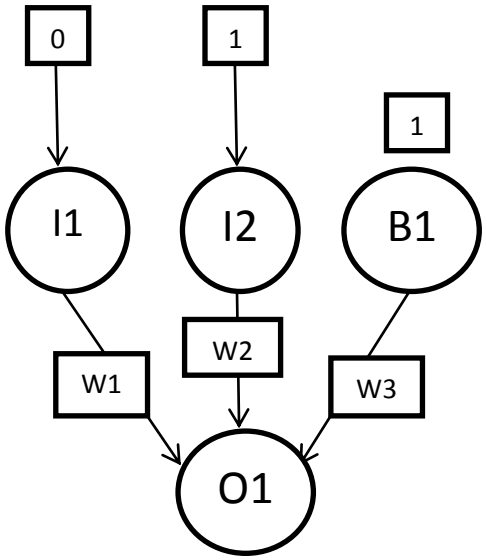
Setiap neuron memiliki sifat aktivasi yang akan mentransformasi sinyal yang masuk ke neuron tersebut sehingga dihasilkan output dari neuron tersebut.

Sebagai contoh sederhana dengan neuron output Y dan neuron input X_1,X_2,X_3 dengan bobot penghubung masing-masing. Nilai sinyal yang masuk adalah X_1,X_2,X_3

pada neuron Y nilai sinyal yang masuk akan dijumlahkan dan menjadi total input (y_in) bagi neron Y, $y_in = W_1X_1+W_2X_2+W_3X_3+b$.

Menurut [19] Backpropagation merupakan sebuah metode sistematik pada jaringan saraf tiruan dengan menggunakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh perceptron dengan banyak layar lapisan untuk mengubah bobot-bobot yang ada pada lapisan tersembunyinya. *Backpropagation* adalah pelatihan jenis terkontrol dimana menggunakan pola penyesuaian bobot untuk mencapai nilai kesalahan yang minimum antara keluaran hasil prediksi dengan keluaran yang nyata. Untuk mendapatkan error jaringan ini, fase *forwardpropagation* harus dilakukan sebelumnya. Saat propagasi dalam arah maju, neuron diaktivasi dengan menggunakan fungsi aktivasi sigmoid.

Rumus aktivasi sigmoid adalah:



Sismoid

$$f(x)=\frac{1}{1+e^{-x}}$$

W1= I1 → O1= 0,5

W1= I2 → O1= 0,6

W1= B1 → O1= 0,7

O1= a ((I1.W1)+(I1.W2)+W3)

O1= a ((0.0,5)+(1.0,6)+0,7)= a (1,3) = 0,79

2.10. Eclipse

Menurut [18] Eclipse adalah sebuah IDE (Integrated Development Environment) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (platform-independent). Berikut ini adalah sifat dari Eclipse:

a. Multi-platform: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
b. Mulit-language: Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
c. Multi-role: Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan open source, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan plug-in. Eclipse dibuat dari kerja sama antara perusahaan-perusahaan anggota 'Eclipse Foundation' (beserta individu-individu lain). Banyak nama besar yang ikut dalam 'Eclipse Foundation', termasuk IBM, BEA, Intel, Nokia, Borland. Eclipse bersaing langsung dengan Netbeans IDE. Plugin tambahan pada Eclipse jauh lebih banyak dan bervariasi dibandingkan IDE lainnya.

1. IntelliJ IDEA (commercial, free 30-day trial)
2. Oracle JDeveloper (free)
3. Xinox JCreator (ada versi berbayar maupun free)

Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak IBM Visual Age for Java 4.0. Produk ini diluncurkan oleh IBM pada tanggal 5 November 2001, yang menginvestasikan sebanyak US$ 40 juta untuk pengembangannya. Semenjak itu konsursium Eclipse Foundation mengambil alih

untuk pengembangan Eclipse lebih lanjut dan pengaturan organisasinya. Sejak versi 3.0, Eclipse pada dasarnya merupakan sebuah kernel, yang mengangkat plug-in. Apa yang dapat digunakan di dalam Eclipse sebenarnya adalah fungsi dari plug-in yang sudah diinstal. Ini merupakan basis dari Eclipse yang dinamakan Rich Client Platform (RCP).Berikut ini adalah komponen yang membentuk RCP:

1. Core platform
2. OSGi
3. SWT (Standard Widget Toolkit)
4. JFace
5. Eclipse Workbench

Secara standar Eclipse selalu dilengkapi dengan JDT (Java Development Tools), plug-in yang membuat Eclipse kompatibel untuk mengembangkan program Java, dan PDE (Plug-in Development Environment) untuk mengembangkan plug-in baru. Eclipse beserta plug-in-nya diimplementasikan dalam bahasa pemrograman Java. Konsep Eclipse adalah IDE yang terbuka (open), mudah diperluas (extensible) untuk apa saja, dan tidak untuk sesuatu yang spesifik. Jadi, Eclipse tidak saja untuk mengembangkan program Java, akan tetapi dapat digunakan untuk berbagai macam keperluan, cukup dengan menginstal plug-in yang dibutuhkan. Apabila ingin mengembangkan program C/C++ terdapat plug-in CDT (C/C++ Development Tools). Selain itu, pengembangan secara visual bukan hal yang tidak mungkin oleh Eclipse, plug-in UML2 tersedia untuk membuat Diagram UML. Dengan menggunakan PDE setiap orang bisa membuat plug-in sesuai dengan keinginannya. Salah satu situs yang menawarkan plug-in secara gratis seperti Eclipse downloads by project.

Sejak tahun 2006, Eclipse Foundation mengkoordinasikan peluncuran Eclipse secara rutin dan simultan yang dikenal dengan nama Simultaneous Release. Setiap versi peluncuran terdiri dari Eclipse Platform dan juga sejumlah proyek yang terlibat dalam proyek Eclipse. Tujuan dari sistem ini adalah untuk menyediakan distribusi Eclipse dengan fitur-fitur dan versi yang terstandarisasi. Hal ini juga dimaksudkan

untuk mempermudah Deployment dan maintenance untuk sistem enterprise, serta untuk kenyamanan.Peluncuran simultan dijadwalkan pada bulan Juni setiap tahunnya.

2.11. Open Source Computer Visio (Open CV)

Open Source Computer Vision adalah library open source multi-platform yang berlisensi Berkeley Software Distribution (BSD) yang bersifat open source sehingga bebas digunakan untuk hal-hal yang bersifat akademis dan penelitian. Library ini berfungsi untuk mentransformasikan data dari citra diam atau kamera video ke salah satu keputusan atau representasi baru. Semua representasi tersebut dilakukan untuk mencapai beberapa tujuan tertentu.



Gambar 2.12 OpenCV

OpenCV dikembangkan oleh Visual Interactivity Group di Intel's Microprocessor Research Lab pada tahun 1999. Tujuan dikembangkannya OpenCV adalah untuk membangun suatu komunitas open source dan menyediakan situs yang mendistribusikan upaya dari komunitas sehingga dapat dikonsolidasi dan dioptimalkan dari segi kinerjanya.

Telah banyak penambahan kelas OpenCV, sehingga OpenCV sampai saat ini telah memiliki lebih dari 500 fungsi. Berikut ini beberapa fungsi yang dimiliki oleh OpenCV :

1. Metoda Geometrik
2. Pengenalan Objek
3. Pengukuran
4. Segmentasi
5. *Utilitas*
6. Fitur
7. Transformasi Citra
8. Piramida Citra
9. *Tracking*
10. Kalibrasi kamera
11. Pengolahan citra

Alasan penggunaan OpenCV karena kostumisasi yang dimiliki oleh OpenCV yang tinggi dan kinerja yang baik pada program berbasis bahasa native C/C++. Selain itu juga terdapat beberapa kelebihan OpenCV, diantaranya :

1. OpenCV gratis, karena open source library.
2. Memliki dokumen yang cukup banyak.
3. Dapat bekerja lebih cepat pada presesor berbasis Intel.

Selain kelebihan, OpenCV juga memiliki kekurangan, yaitu OpenCV kurang mendukung bahasa berbasis .NET seperti C#, Visual Basic dan lainnya, sehingga dibutuhkan wrappers mendukungnya. Pada penelitian ini menggunakan OpenCV, karena tools yang dimiliki oleh OpenCV dapat mendukung untuk penelitian deteksi makanan dan obat menggunakan metoda String Matching ini.

2.12. JavaScript

JavaScript awalnya bernama LiveScript yang dikembangkan pertama kali pada tahun 1995 oleh Brendan Eich dari Netscape Communications. LiveScript berfungsi

sebagai bahasa sederhana untuk browser Netscape Navigator 2 dan mendapatkan banyak kritik karena dianggap kurang aman karena pengembangannya yang terkesan buru-buru dan tidak ada pesan kesalahan yang ditampilkan setiap terjadi kesalahan pada saat menyusun kode program. Kemudian Netscape berkolaborasi dengan Sun Microsystem seorang pengembang bahasa pemrograman Java diakhir tahun 1995 dan mengubah nama LiveScript menjadi JavaScript.

Menurut Negrino dan Smith (2001), JavaScript adalah sebuah bahasa pemrograman yang bisa digunakan untuk menambahkan interaktifitas pada halaman web. JavaScript merupakan bahasa scripting (bahasa pemrograman yang ringan) yang popular di internet dan berisi baris kode yang dijalankan di komputer menggunakan program browser yang mendukung seperti Google Chrome, Internet Explorer (IE), Mozilla Firefox, Netscape dan Opera. JavaScript disisipkan dalam halaman HTML dengan diapit menggunakan tag Script. JavaScript dijalankan di komputer user, sehingga proses tidak perlu dilakukan pada server. Dengan adanya JavaScript, kemampuan sebuah file dokumen HTML akan menjadi semakin luas dan kuat. Selain itu, JavaScript juga merupakan bahasa interpreter yang berarti skrip dieksekusi tanpa proses kompilasi. JavaScript didesain untuk menambah interaktif suatu web.

2.13. UML

UML (Unified Modeling Language) adalah sekumpulan pemodelan konvensi yang digunakan untuk menetukan atau menggambarkan sebuah sistem perangkat lunak dalam kaitannya dengan objek [7]. UML juga diartikan sebuah bahasa grafik standar yang digunakan untuk memodelkan perangkat lunak berbasis objek. UML pertama kali dikembangkan pada pertengahan tahun 1990an dengan kerjasama antara James Rumbaugh, Grady Booch dan Ivan Jacobson, yang masing- masing telah mengembangkan notasi mereka sendiri di awal Tahun 1990an. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan system

yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

1. Analisis dan desain Berorientasi Objek

Pemrograman berorientasi objek bekerja dengan baik ketika bersamaan dengan *Object-oriented Analysis and Design Process (OOAD).*[8] mengatakan jika kita membuat program berorientasi objek tanpaOOAD, ibarat membangun rumah tanpa terlebih dahulu menganalisa apa saja yang dibutukan oleh rumah itu, tanpa perencanaan, tanpa *blueprint*, tanpa menganalisis ruangan apa saja yang diperlukan, berapa besar rumah yang akan dibangun dan sebagainya.

a. Objek

Orientasi objek merupakan teknik dalam menyelesaikan masalah yang kerap muncul dalam pengembangan perangkat lunak. Teknik ini merupakan titik dalam menentukan cara yang efektif dalam pembangunan system dengan metode yang paling banyak dipaki oleh para pengembang perangkat lunak saat ini. Orientasi objek merupakan teknik pemodelan sistem riil yang berbasis objek. Intinya dari konsep ini adalah objek yang merupakan model daris sitem nyata. Menurut [9] bahwa objek adalah entitas, karakter *(behavour)* dan kadangkala disertai kondisi *(state).*

b. Kelas

Kelas adalah penggambar satu set objek yang memiliki atribut dan behavior yang sama. Kelas mirip tipe data pada program non objek, tapi lebih komprehensif karena terdapat struktur sekaligus karakteristiknya. Kita dapat membentuk kelas baru yang lebih spesifik dari kelas generalnya. Kelas dan objek merupakan jantung dari pemrograman berorientasi objek.untuk menghasilkan program ini sangat penting untuk selalu berfikirdalam konsep objek.

2. Evolusi UML

[10] menjelaskan bahwa sebelumnya ada UML, para pengembang bahasa pemrograman berorientasi objek sulit unutuk berkomunikasi satu sama lain. Kira-kira memiliki 50 jenis notasi dan grafik yang menggambarkan bahasa pemrograman berorientasi objek pada waktu itu. Pada pengguna yang berlainan ini saling berebut pengaruh agar notasi yang digunakan menjadi notasi standar. Walaupun dijumpai beberapa notasi yang cukup jelas dan sangat cocok untuk menyelesaikan problem dalam pembuatan perangkat lunak *Object Oriented Program*(OOP).

Pada bulan Oktober 1994, Jim Rumbaugh, penemu notasi *Object Modelling Technique (OMT)* dan Graddy Booch, penemu *Booch Method* (Metode Booch) bersama menyamakn notasi mereka. Di tahun yang sama, Ivar Jacobson ikut bergabung. ketiga orang itu sering disebut sebagai "*three Amigos*". Bersama membangun notasi standar OOP untuk software *Rational IBM*.

Ketiga pengembang UML tersebut telah mengalami kesulitan akibat kompleksnya permasalahan OOP yang ada, namun mereka dibantu oleh *Object Management Group (OMG)*. Artinya gabungan dari kurang lebih 800 perusahaan pengembang perangkat lunak berorientasi objek.

Maka sejak tahun 1997, divisi *Revision Task Force (RTF)* milik OMG beberapa kali merevisi UML. Revisi dimaksudkan agar memperkuat konsistensi notasi, meningkatkan kekompakan antar user dan pengembang perangkat lunak. Akan tetapi UML akhirnya mengikuti perkembangan *software-software* bebasis objek yang ada (*Java*) dari sisi pendekatan komponen dan kemampuan *tools software* tersebut. Setelah dilakukan perubahan secara sistematik, akhirnya dihasilkan UML 2.0 pada tahun 2003.

3. Komponen UML

Bagian-bagian atau Komponen utama dari UML adalah view, diagram, model element, dan *general mechanism*.

a. View

View digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. View bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram. Beberapa jenis view dalam UML antara lain: use case view, logical view, component view, concurrency view,dan deployment view.

(a). Use case view

Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan external aktors. Aktor yang berinteraksi dengan sistem dapat berupa user atau sistem lainnya. View ini digambarkan dalam use case diagrams dan kadang-kadang dengan activity diagrams. View ini digunakan terutama untuk pelanggan, perancang (designer), pengembang (developer), dan penguji sistem (tester).

(b). Logical view

Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis (class, object,dan relationship) dan kolaborasi dinamis yang terjadi ketika object mengirim pesan ke object lain dalam suatu fungsi tertentu.View ini digambarkan dalam class diagrams untuk struktur statis dan dalam state, sequence, collaboration, dan activity diagram untuk model dinamisnya. View ini digunakan untuk perancang (designer) dan pengembang (developer).

(c). Component view

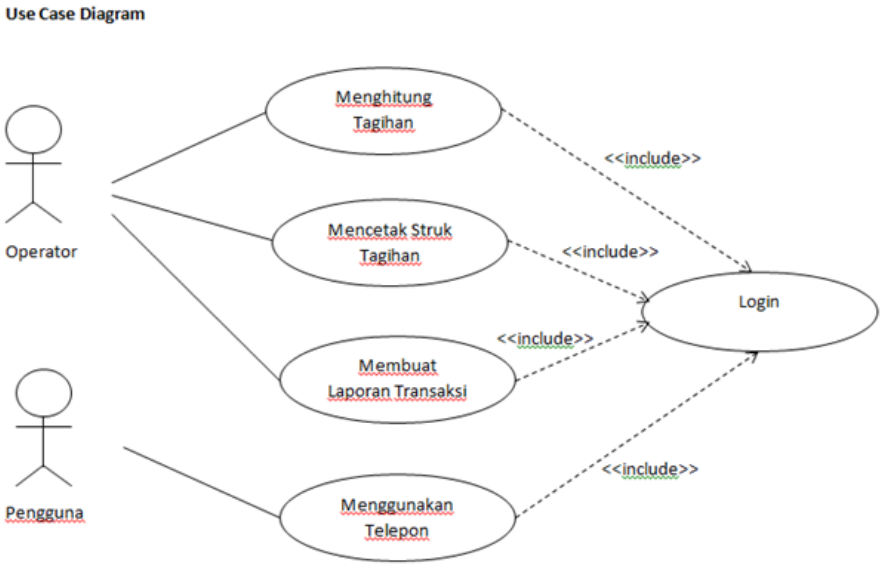
Mendeskripsikan implementasi dan ketergantungan modul. Komponen yang merupakan tipe lainnya dari code module diperlihatkan dengan struktur dan ketergantungannya juga alokasi sumber daya komponen dan informasi administrative lainnya. View ini digambarkan dalam component view dan digunakan untuk pengembang (developer).

2.14. Use Case Diagram

Menurut [11] use case menggambarkan fungsi tertentu dalam suatu sistem berupa komponen, kejadian atau kelas. Sedangkan [7] mengartikan use case sebagai urutan langkah-langkah yang secara tindakan saling terkait (skenario), baik terotomasisasi maupun secara manual, untuk tujuan bisnis tunggal. Salah satu kontributor terhadap diagram use case dalam UML adalah Ivar Jacobsen. Use case menggambarkan external view dari sistem yang akan kita buat modelnya [12] mengatakan bahwa model use case, tetapi yang perlu di ingat, diagram tidak identik dengan model karena model lebih luas dari diagram. Komponen-komponen pembentuk use case adalah Aktor, use case dan hubungan *(link).* Pada diagram use case, relasi digambarkan sebagai sebuah garis antara symbol. Pemaknaan relasi berbeda-beda tergantung bagaiman garis tersebut digambarkan dan tipe symbol apa yang digunakan untuk menghubungkan garis tersebut. Relasi yang digunakan UML 2.0 adalah generalisasi , inklusi dan ekstensi.

2.14.1. Aktor

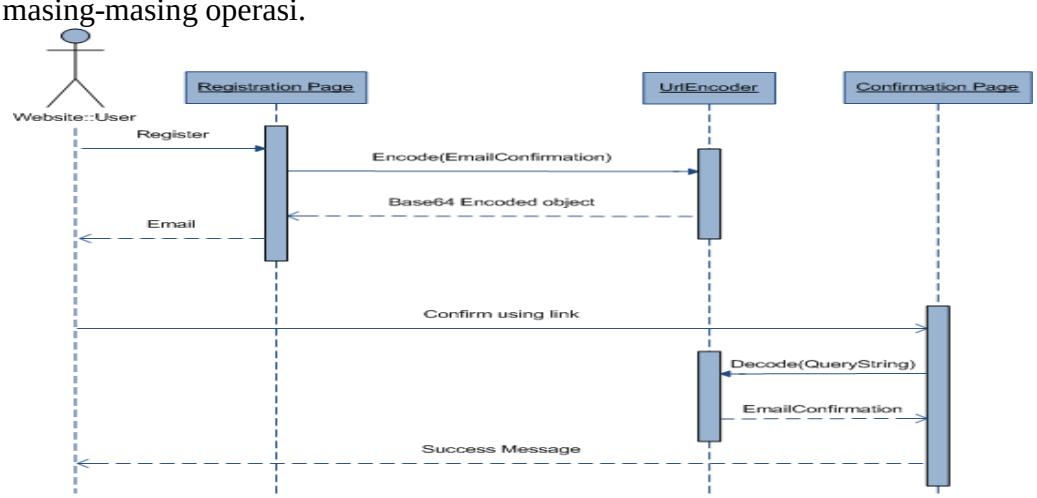
[10] Menyarankan sebelum membuat *use case* dan menentukan aktornya, agar mengidentifikasikan siapa saja yang terlibat dalam sistem kita. Pihak yang terlibat biasanya dinamakan *stakeholder*, langkah awal yang baik adalah mempertimbangkan kebutuhan klien dan pelanggan dalam membentuk *use case*. Setiap sistem memiliki *stakeholder* potensial yang harus dipertimbangkan karena berpengaruh terhadap kinerja sistem tersebut. Bahkan untuk beberapa jenis sistem tertentu bisa saja *hacker* masuk ke dalam kategori *stakeholder* karena berpengaruh terhadap sistem. Selama proses mencari kebutuhan sistem, hampir sebagian besar waktu dihabiskan untuk menentukan aktor yang dipilih dari *stakeholder* hasil identifikasi awal.



Gambar 2.13 Diagram Use Case

2.15. Sequence Digram

Sequence diagram menjelaskan secara detil urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case. Interaksi yang terjadi antar class, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.



Gambar 2.14 Sequence Diagram

2.16. Class Diagram

Class adalah suatu diagram yang memperlihatkan atau menampilkan struktur darisebuah sistem, sistem tersebut akan menampilkan sistem kelas, atribut dan hubungan antara kelas ketika suatu sistem telah selesai membuat diagram. *Class diagram* menunjukkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.

Diagram kelas merupakan kumpulan kelas-kelas objek, oleh karena itu pengertian kelas sangat penting sebelum merancang diagram kelas. [4] arti dari kelas sebagai satu set objek yang memiliki atribut dan perilaku yang sama. Kelas kadang-kadang disebut juga kelas objek *(object class).* Objek yang berupa buku analisis disain dan buku pemrograman terstruktur kita kelompokan dalam satu kelas, yaitu kelas buku. Kedua objek memiliki atribut dan perilaku yang serupa. Secara teknis, [10] mengartika sebuah kelas sebagai suatu definisi sumber daya yang termasuk di dalam informasi-informasi yang menggambarkan fitur suatu entitas dan bagaimana penggunaannya. Sedangkan objek adalah entitas yang bersifat unik yang mengikuti aturan-aturan yang sudah didefinisikan oleh kelasnya.

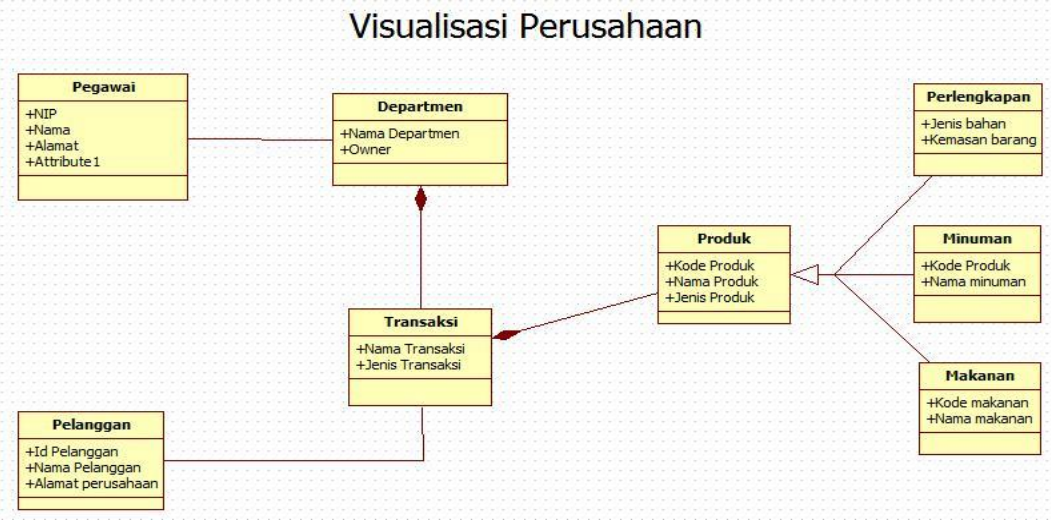
Atribut merupakan suatu kelas yang isinya adalah point-point seperti warna mobil, jumlah sisi bentuk yang disebut sebagai atribut. Atribut bisa saja sederhana (integer, floating-point) bisa juga kompleks. Atribut digambarkan dalam dua notasi yang berbeda yaitu inline atau hubungan antarkelas.

Atribut inline adalah atribut yang dapat menulis atribut kelas disebelah kanan notasi kotak. Tidak ada perbedaan semantic atau atribut inline dan atribut relasi. Keduanya bermaksud merincikan kelas yang akan dibuat.

Atribut relasi adalah atribut yang bisa mempresentasikan sebuah atribut dengan menghubungkan kelas lainnya. Untuk itu kita menggunakan notasi relasi sebagai sarana pembentuk atribut relasi dalam diagram kelas yang sangat besar. Dengan cara itu dapat dibuat atribut yang lebih detail dan kompleks.

Atribut turunan adalah atribut yang diberi notasi slash "/". Symbol ini bermaksud memberi informasi kepada inplementator bahwa atribut tersebut tidak terlalu penting.

Atribut multiplicity merupakan atribut yang menyatakan berapa banyak isi dari atribut yang terbentuk saat kelas diinisialisasi. Misalnya, kelas mobil akan memiliki 4 roda jadi multiplicity atribut roda adalah 4.



Gambar 2.15 Class Diagram

2.17. Perangkat Mobile

Hadirnya perangkat Mobile (ponsel) pada awalnya hanya sebagai sarana berkomunikasi dengan kelebihan yang bisa dibawa kemana- mana. Namun sekarang telah berubah menjadi suatu peralatan yang sangat penting. Dengan adanya internet di dalam ponsel yang telah menjadi kekuatan yang luar biasa. Sehingga dapat diibaratkan ponsel menjadi suatu magnet yang sangat penting bagi para penggunanya.bahkan di dalam ponsel saja sudah banyak fitur yang digunakan sama seperti komputer. Sehingga dengan situasi seperti ini google memanfaatkannya dengan meluncurkan ponsel berbasis android sebagai salah satu andalannya saat ini.

Hingga sampai saat ini google berhasil membuat persaingan yang terjadi pada dunia mobile dari segi Sistem Operasi hingga aplikasi- aplikasi pendukungnya.

2.18. Multimedia

Multimedia berasal dari kata Multi dan Media. Multi berarti banyak atau lebih dari satu, dan media dapat diartikan sebagai saranayang dipakai untuk menyampaikan berbagai informasi. Multimedia dapat dikatakan juga sebagai suatu bentuk baru untuk menggabungkan lebih dari satu media, yang di dalamnya terdapat elemen- elemen teks, gambar suara dan animasi. Meskipun hanya beberapa aplikasi, multimedia yang hanya mengandung dua elemen di atas pun dapat dikatakan sebagai multimedia. Seorang yang merancang dan dan membentuk suatu aplikasi multimedia disebut multimedia developer. Bila dalam suatu aplikasi multimedia pengguna diberi suatu kemampuan untuk mengontrol elemen-elemen yang ada, maka multimedia disebut juga intraktif multimedia. Sebuah interaktif multimedia dapat menjadi hypermedia apabila dalam aplikasi multimedia tersebut disediakan struktur dari elemen yang terhubung yang dapat digunakan oleh pengguna. Dengan menggunakan multimedia informasi dapat diterima dengan kedua indera penglihatan dan pendengaran mendeteksi bentuk aslinya. Berbeda apabila informasi tersebut hanya disampaikan dalam bentuk teks (angka dan huruf) maka lahirnya multimedia merupakan perpaduan dari hasil kemajuan teknologi elektronik, komputer, dan perangkat lunak

2.19. Android

Android adalah sebuah sistem operasi untuk berbagai perangkat mobile seperti handphone, netbook dan komputer tablet. Sistem operasi ni dikembangkan oleh google dengan berbasis kernel Linux versi 2.6 dan berbagai perangkat lunak dari GNU yang bersifat open source. Pada awalnya, android dikembangkan oleh perusahaan android Inc, namun kemudian perusahaan tersebut diakuisisi oleh google sehingga menjadi produk google. Sekarang ini pengembangan android ditentukan

oleh sebuah konsorsium bernama *Open Handset Alliance (OHA)* yang terdiri atas berbagai vendor perangkat mobile, Komputer, dan telekomunikasi seperti Intel, Nvidia, Google, Samsung, Sprint, T-Mobile, Motorola, LG, Sony Ericsson, Toshiba, Vodafone, serta masih banyak yang lain dan anggotanya terus bertambah. Karena android bersifat Open Source dan mempunyai lisensi apache yang sangat terbuka dan bebas, maka Android menjadi sistem operasi yang sangat popular bagi berbagai produsen perangkat mobile. Berdasarkan persentasenya android mencapai 1,1 Miliar di tahun 2014 [1]. Salah satu keunggulan android adalah adanya komunitas para developer dan programmer yang luas untuk mengembangkan berbagai aplikasi yang berjalan diperangkat berbasis Android sehingga mampu memperluas fitur dan kemampuan perangkat tersebut. Saat ini sudah terdapat lebih dari 70000 aplikasi android sehingga berada di peringkat ke dua di dunia. Android relatif mudah dan cepat karena menggunakan bahasa pemrograman Java dengan dukungan library Java yang dikembangkan oleh google.