BAB 2
TINJAUAN PUSTAKA

2.1 Computer Vision

Sedangkan *Computer Vision* dapat didefinisikan setara dengan pengertian pengolahan citra yang dikaitkan dengan akuisisi citra, pemrosesan, klasifikasi, pengakuan, dan pencakupan keseluruhan, pengambilan keputusan yang diikuti dengan pengidentifikasian citra.

Ballard dan Brown (1982) mendefinisikan *Compueter Vision* sebagai suatu kegiatan awal pengotomatisan dan pengintegrasian suatu pemrosesan dan representasi sebagai suatu persepsi visual dengan tahap-tahap tertentu.

Niblack (1986) memaparkan pengolahan citra sebagai "pengolahan citra dengan komputer", dan kemudian definisi singkat ini ditambah dengan "dan output dan hasil pengolahan citra tersebut juga berupa citra." Sementara "*Computer Vision* melibatkan banyak teknik dari pengolahan citra dan juga lebih luas lagi, dalam wilayah suatu sistem yang lengkap, "mesin melihat".

Boyle dan Thomas (1988) memberi pengertian bahwa *Computer Vision* lebih dari hanya sekedar *image recognition*. Mereka juga menghadirkan operasi *low level processing* sebagai suatu algoritma pengolahan citra yang dapat kita sebut *purely* yang kemudian mengkategorikan citra tersebut didalam suatu Computer Vision.

Jensen (1986) memberi judul untuk bukunya Introductory Digital Image Processing-a remote sensing perspective, dan memasukan bagian-bagian subtansial berdasarkan image acquisition dan region classification.

Gonzales dan Wintz (1977) memasukan bagian-bagian substansial (Jensen) berdasarkan deskripsi region dan deskripsi relasional, dan keseluruhan pendapat yang berkaitan pegelompokan bagian-bagian tersebut di dalam wilayah *Computer Vision* [4].

Aplikasi dari pengolahan citra dan *Compuer Vision* dalam berbagai bidang atau disiplin ilmu pengetahuan, teknologi, dan kehidupansehari-hari, dirangkum di dalam tabel berikut [4]:

Tabel 2.1 Aplikasi Computer Vision Dan Pengolahan Citra

Nama Aplikasi	Nama Aplikasi
Pictorial database	Qualitiy control of food
Graphics Design	Bacterial growth scanning
Engraving	Geological modelling
Textile design	Autonomous navigation
Cartography	Cosmetics
Metallurgy	Astronomy
Materials science	Defence application
Medical imaging	Particle tracking
Object recognition	Fishing using ultrasound
Microscopy	Pottery blemish identification
Satellite picture processing	Eye retina comparison
Particle counting and sizing	Fluid mechanics
Typesetting	Journalism – picture enhancement
3D reconstruction	Security – picture enhancement and identification
Photograpic security	Weather mapping
Fingerprint matching	Document transfer
Document reading	Document comparison

2.1.1 The Human Vision System

Beberapa deskirpsi yang berkaitan dengan *Computer Vision* telah memaparkan seberapa jauh kemampuan visi komputer mendekati kemampuan visi manusia.

Bila dikaitkan dengan apa-apa yang ada di dalam *Computer Vision,* sepasang mata manusia masih lebih unggul dibandingkan segala jenis kamera tercanggih manapun yang pernah dibuat manusia. Retina misalnya, yang menangkap suatu citra secara terbalik, berisi dua kelas reseptor pencahayaan diskrit – bentuk kerucut dan balok. Ada sekitar 6 sampai 7 juta kerucut didalam kelopak mata, dan kebanyakan dari mereka berada pada bagian pusat retina, yag disebut *fovea*. Kerucut-kerucut ini sangat sensitif terhadap warna-warna cerah (*bright*), dan tidak mampu beroperasi pada sumber cahaya yang suram (*dim*). Masing-masing kerucut dihubungkan ke otak manusia oleh urat-urat syarafnya. Ditaksirkan paling sedikit ada sekitar 75 juta balok di dalam mata yang berada

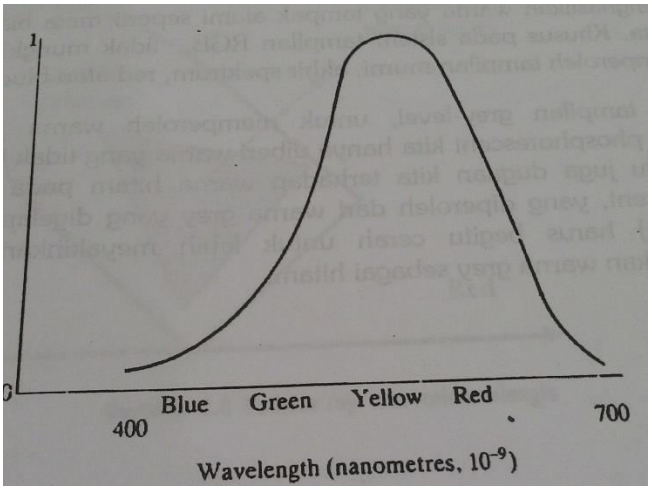
pada permukaan retina. Mereka sensitif terhadap intensitas cahaya yang bukan warna. Mereka bekerja sama dengan syaraf-syaraf pada akhirnya dan hanya melayani untuk memberikan citra keseluruhan yang ditangkap oleh mata. Mampu bekerja pada kondisi cahaya suram, hanya sebagai suatu citra *monochrome, citra multi-grey-level* yang diperoleh dalam situasi suram [4].

Range intensitas dimana mata mampu beradaptasi dengan cahaya adalah sekitar 10^{10}, dari cahaya yang dapat ditangkap oleh mata terendah hingga cahaya yang sangat menyilaukan. Secara praktis, kemampuan mata yang menakjubkan ini dilakukan dengan mengubah sesitivitas yang dimiliki oleh mata yang biasanya tergantung pada level kecerahan ruang. Walaupun mata mampu mengatasi keadaan jangkauan tingkatan cahaya yang begitu lebar, namun demikian ia tidak mampu membedakan secara serentak tingkatan perbedaan suramdan tingkatan perbedaan cerah [4].

Didalam tampilan *grey-level*, untuk memperoleh warna putih pada permukaan *phosporescent* kita hanya diberi warna yang tidak benar-benar putih. Begitu juga dugaan kita terhadap warna hitam pada permukaan *phosporescent*, yang diperoleh dari warna grey yang digelapkan. Warna putih (*white*) harus begitu cerah untuk lebih meyakinkan mata kita mempersepssikan warna grey sebgai warna hitam [4].

2.1.2 Persepsi Mata Terhadap Warna

Persepsi kecerahan yang dilakukan oleh mata merupakan suatu fungsi panjang gelombang dan dipaparkan oleh suatu fungsi luminositas relatif, fungsi luminositas relatif dapt dilihat pada gambar berikut :



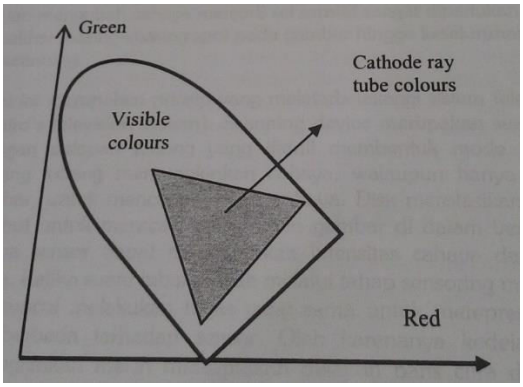
Gambar 2.1 Fungsi Luminositas Relatif

Konsekuensinya, pencaran cahaya pada suatu panjang gelombang di tengah spektrum selalu terlihat lebih cerah daripada pancaran cahaya dengan nilai yang sama tetapi dengan panjang gelombang pad sekitar titik-titik ujung spektrum. Oleh karena itu warna merah dan birulebih mampu menghasilkan kecerahan yangsebanding denga wara hijau dan kuning.

Tampilan warna biasanya menggunakan pola baku “perubahan warna”. Sistem RGB hanya membatasi pada titik-titik red,green, dan blue pada layar monitor. Jika titik-titik tersebut cukup kecil dan pengamat berada dalam jarak yang cukup jauh maka hal itu akan memberikan suatu visual sebagai satu piksel warna yang merupakan salah satu dari ketiga warna utama diatas. Ketiga warna utama tersebut bila dikombinasikan dengan dua warna utama sesamanya maka tidak akan menghasilkan warna utama ketiga. Boleh kita katakan bahwa ketiga warna tersebut merupakan warna independen.

Tampilan yang menggunakan warna tambahan akan memberdayakan ketiga warna tersebut untuk menghasilkan warna putih cerah. Contoh, pencampuran ketiga warna tersebt akan menghasilkan warna hitam pada praktisnya, titik-titik *phosporescent* pada tampilan layar monitortidak memberikan masing-masing suatu nilaipanjang gelombang tunggal yang mewakili warna, tetapi akan menghasilkansuatu range panjang gelombang yang berpusat pada warna yang akan dihasilkan. Hal ini memberikan masalah khusus, didalam sistem tampilan

CRT sangat tidak mungkin untuk menghasilkan warna yang tampak alami seperti mata biasamelihat alam nyata. Khusus pada sistem tampilan RGB, tidak mungkin bagi kita untuk memperoleh tampilan murni, akhir spektrum, red atau blue.



Gambar 2.2 *Chatode Ray Tube Color Triangle*

2.2 Video

Video merupakan sinyal elektronik yang mewakilkan gambar bergerak. Penggunaan umum video adalah televisi namunvideo juga digunakan dalam aplikasi teknik, keilmuan, produksi, dan keamanan. Terdapat dua kategori video, yatu video analog dan video digital [1].

2.2.1 Video Analog

Video analog mengodekan informasi gambar dengan memvariasikan voltase dan atau frekuansi sinyal. Terdapat dua format pada video analog, yaitu format elektrik dan format kaset. Format pengodean pada video analog adalah sebagai berikut:

1. NTSC
2. PAL
3. SCAM

Sedangkan format elektrik pada video analog adalah sebagai berikut :

1. RF
2. Composite video
3. Component video
4. S-Video

5. RGB

Format kaset pada video analog adalah sebagai berikut :

1. Ampex
2. VERA (BBC)
3. U-matic (Sony)
4. Betamax (Sony)
5. Betacam
6. Betacam SP
7. VHS (JVC)
8. S-VHS (JVC)
9. VHS-C (JVC)
10. Video 2000 (Philips)
11. 8mm tape

2.2.2 *Video Digital*

Video digital dapat disebut array tiga dimensi dari piksel berwarna. Array tersebut terdiri dari dua dimensi melayani arah spasial dari gambar bergerak (horizontal dan vertikal) dan satu dimensi lainnya akan merepresentasikan domain waktu.

Arsitektur video digital tersusun atas sebuah format untuk mengodekan dan memainkan kembali file video dengan komputer dan menyertakan sebuah pemutar (*player*) yang mengenali dan membuka file yang dibuat untuk format tersebut. Contoh arsitektur video digital diantaranya adalah :

1. Apple QuickTime
2. Microsoft Windows Media Format
3. Real Network RealMedia

Adapun format video terkait dengan arsitektur tersebut adalah :

1. QuickTime movie (.mov)
2. Audio Video Interleaved (.avi)
3. RealMedia (.rm)

Berikut ini merupakan video digital dalam format kaset :

1. D1 (Sony)

2. D2 (Sony)
3. D3
4. D4
5. D5
6. Digital Betacam (Sony)
7. Betacam IMX (Sony)
8. D-VHS (JVC)
9. DV
10. MiniDV
11. MicroMV
12. Digital8 (Sony)

Video digital terdiri atas serangkaian gambar digital yang ditampilkan pada kecepatan konstan, gambar tersebut dinamakan *frame* dan satuan ukuran untuk menghitung *frame* rata-rata yang ditampilkan disebut *frame per second* (fps). Setiap *frame* merupakan gambar digital yang terdiri dari raster piksel dan akan mempunyai lebar sebanyak W piksel dan tinggi sebanyak H piksel, maka dapat dikatakan bahwa *frame size* adalah W x H.

Warna merupakan satu properti yang dimiliki piksel. Warna piksel direpresentasikan dengan jumlah bit yang tetap. Semakin banyak bit, semakin banyak variasi warna yang dapat dihasilkan yang disebut *color depth* (CD) dari video. Sebagai contoh, sebuah video mempunyai durasi (T) 1 jam (3.600 sec), *frame size* 640 x 480 (W x H) dengan *color depth* 24 bit, dan *frame rate* 25 fps akan mempunyai properti sebagai berikut :

Pixel per frrame = 640 × 480 = 307.200
Bit per frame = 307.200 × 24 = 7.372.800 = 7,37 Mbit
Bit rate (BR) = 7,37 * 25 = 184,25 Mbit/sec
Video size (VS) = 184 Mbit/sec × 3600 sec = 662.400 Mbit
= 82.800 Mbyte = 82,8 Gbyte

Hal penting dariproperti adalah bitrate dan *video size* dengan rumus keduanya yang terkait dengan semua properti adalah sebagai berikut :

Pixels_per_frame = W × H

*Pixels*_per_*second* = W × H × FPS

bits_per_*frame* = W × H × CD

Dari perhitungan diatas, pendigitalan dan penyimpanan klip video *full-motion* 640 x 480 px selama 1 jam ke dalam komputer, dengan *color depth* 24 bit, dan *frame rate*-nya 25 fps memerlukan komputer yang dapat mengirimkan data kurang lebih 184 Mbit/sec. Artinya, komputer harus mempunyai kecepatan tinggi dan berkapasitas besar. Suatu skema kompresi video digital yang disebut codec (coder/decoder) akan sangat membantu dalam penghematan pada hal ini [1].

2.2.3 Media Video Digital

CD-ROM menyediakan media distribusi yang baik untuk video digital karean tidak mengeluarkan biaya yang mahal untuk produksi massal dan dapat menyimpan kuantitas informasi dalam jumlah besar. Selain CD-ROM dapat menggunakan DVD atau Blu-Ray yang dapat menjadi media penyimpanan video digital.

Bentuk penyimpanan dari CD-ROM, DVD, maupun Blu-Ray yaitu berbentuk *disc* (cakram). Selain bentuk cakram terdapat juga media penyimpana video digital dalam bentuk *tape* (pita).

Berikut format pada media penyimpanan video digital berbentuk pita :

1. Betacam, Betacam SP, Betacam SX, Betacam IMX, Digital Betacam atau DigiBeta

   Format tersebut dikembangkan oleh Sony yang berbasis teknologi Betamax
2. HDCAM

   Sony memperkenalkan format ini sebagai alternatif *high-definition* untuk DigiBeta
3. D1, D2, D3, D4, D5, D9 (Digital-S)

   Format ini merupakan beragam standar video digital komersial dari SMPTE
4. DV, miniDV

   DV dan miniDV saat ini digunakan untuk *camcorder* berbasis *video tape* yang dirancang untuk hasil yang berkualitas tinggi dan mudah diedit.

Format ini juga merekam high-definition video (HDV) dalma format MPEG-2

5. DVCAM,DVPRO

   DVCAM dan DVDPRO digunakan dalam operasi penyiaran profesional. Keduanya mirip dengan DV, tetapi secara umum mempunyai daya tahan yang lebih tinggi dan mempunyai kemudahan dalam penanganan audio
6. DVCPRO50, DVCPROHD

   DVCPRO50 dan DVCPROHD mendukung *bandwith* yang lebih tinggi sebagai perbandingan dari DVCPRO milik Panasonic
7. Digital 8

   Digital 8 merupakan format data DV yang direkam pada Hi8 kaset yang kompetibel
8. MicroMV

   MicroMV adalah format data MPEG-2 yang direkam pada tape yang mirip dengan S-VHS [1].

2.3 Pengolahan Citra

Pengolahan citra adalah cabang ilmu informatika untuk memperbaiki kualitas citra agar kualitasnya lebih baik atau lebih mudah diinterpretasi oleh manusia atau komputer. *Input* dan *ouput* dari program pengolahan citra adalah citra [5].

Terdapat dua prinsip daerah aplikasi pengolahan cira digital: peningkatan informasi piktorial untuk interpretasi manusia dan pengolahan citra untuk penyampaian, transmisi dan representasi bagi peralatan persepsi (*perception*).

Sebuah citra dapat didefinisikan sebagai fungsi dua dimensi f(x,y), dimana x dan y adalahh koordinat spasial, dan amplitudo dari f pada sembarang pasang koordinat (x,y) disebut intensitas (*intensity*) atau *gray level* (level keabuan) dari citra pada titik tersebut. Ketika x, y dan nilai intensitas dari f adalah semua terbatas, *discrete quantities*, citra tersebut dinamakan citra digital (*digital image*). Citra digital terdiri dari sejumlah elemen tertentu, setiap elemen mempunyai

lokasi dan nilai tertentu. Elemen-elemen ini disebut *picture element, image element,* dan piksel. Piksel adalah istilah yang sudah digunakan secara luas untuk menyatakan elemen citra digital.

Penglihatan adalah indera yang paling peka sehingga tidak mengejutkan bila citra memainkan peran paling penting dalam persepsi manusia. Bagaimanapun, tidak seperti manusia yang terbatas dalam band penglihatan spektrum elektromanetik (EM), mesin pencitraan mencakup hampir semua spektrum EM, dengna jangkauan mulai dari sinar gamma sampai gelombang radio. Mesin tersebut dapat mengoperasikan citra yang dihasilkan oleh sumber yang manusia tidak bisa hubungkan dengan citra, termasuk ultrasound, *electron microscopy,* dan komputer pembuat citra. Karena itu, pengolahan citra digital meliputidaerah aplikasi yang luas dan bermacam-macam [6].

2.3.1 Sejarah Pengolahan Citra Digital

Satu dari aplikasi citra digital yang pertama adalah industri surat kabar, ketika citra pertama kali diikirimdengna kabel kapal selam antara London dan New York. Pengenalan sistem transmisi kabel laut Bartlane pada awal tahun 1920 mengurangi waktu yang dibutuhkan untuk mentransmisikan citra melintas Atlantik lebih dari satu minggu kurang dari tiga jam. Peralatan cetak khusus mengkodekan citra untuk kabel transmisi dan kemudian merekonstruksinya di tempat penerima.

Beberapa masalah awal dalam peningkatan kualitas visual gamabr digital dulu dihubungkan dengan prosedur pencetakan dan distribusi level intensitas. Metode pencetakan yang digunakan pada saat itu kemudian ditolak pada akhir 1921.

Awalnya sistem Bartlane dapat mengkodekan citra dalam lima perbedaan level keabuan. Kemampuan ini ditingkatkan menjadi level 15 pada 1929. Hasil citra pada saat itu adalah tipikal citra yang didapatkan menggunakan peralatan 15 tone. Sejak periode ini pengenalan sistem untuk mengembangkan *film plate* dengan *light beam* dimodulasikan oleh kode gambar tape yang meningkatkan perhatian terhadap proses reproduksi.

Pengolahan citra digital berkaitan erat dengan perkembangan komputer digital. Nyatanya, citra digital membutuhkan banyak tempat penyimpanan dan kekuatan komputasi. Kemajuan dalam pemrosesan citra digital tergantung pada pengembangan komputer digital dan dukungan teknologi, termasuk *data storage*, *display,* dan transmisi.

Komputer pertama yang mencakup kuat untuk melakukan pekerjaan pengolahan citra digital muncul pada tahun 1960. Kelahiran dari apa yang sekaran gini disebut pengolahan citra digital dapat dilacak pada ketersediaan mesin-mesin tersebut dan pada sekumpulan *sapce program* selama periode tersebut. Kombinasi diambil dari dua pengembangan untuk membawanya fokus pada potensi konsep pengolahan citra digital. Bekerja menggunakan teknik komputer untuk meningkatkan citra dari sebuah tempat penelitian dimulai oleh Jet Propulsion Laboratory (Pasadena, California) pada tahun 1964 ketika gambar bulan ditransmisikan oleh Ranger 7 yang kemudian diproses oleh komputer untuk menyempurnakan bermacam-macam jenis citranya.

Sejak tahun 1960 hingga sekarang, daerah pengolahan citra telam tumbuh sangat cepat. Perkembangan aplikasi dalam dunia medis dan *space program* membuat teknik pengolahan citra digital digunakan dalam aplikasi bersekala besar. Prosedur komputer yang digunakan untuk menigkatkan kontras atau kode level intensitas ke dalam warna mempermudah interpretasi X-ray (sinar X) dan citra lain yang digunakan dalam dunia industri, medis dan ilmu biologi. Geografer menggunakan cara yang sama untuk mempelajari pola polusi. Prosedur peningkatan dan perbaikan citra digunakan untuk memproses citra terdegradasi dari obyek yang tidak dapt dipulihkan atau hasil eksperimen yang mahal untuk diduplikasi. Didalam dunia arkeologi, metode pengolahan citra sukses dalam mengembalikan citra yang kabur (*blurred*). Didalam dunia fisika, teknologi komputer menigkatkan citra eksperimen dalam area seperti *high-energy plasma* dan *electron microscopy*. Kesuksesan aplikasi pengolahan citra digital dapat ditemukan di dunia astronomi, biologi, nuklir, medis, hukum, pertahanan dan industri [6].

2.3.2 Area Penggunaan Pengolahan Citra

Saat ini hampir tidak ada area yang langkah teknisnya tidak dipengaruhi oleh pengolahan citra digital. Area aplikasi pengolahan citra digital ada bermacam-macam. Salah satu cara paling mudah untuk mengembangkan pemahaman dasar tingkat aplikasi pengolahan citra digital adalah dengan mengklasifikasikan menurut sumbernya (misalnya visul, X-ray, dsb). Sumber energi citra yang digunakan saat ini adalah energi spektrum elektromagnetik. Sumber penting yang lain dari energi adalah akustik, ultrasonik dan elektronik. Citra sintetik digunakan untuk pemodelan dan visualisasi yang dihasilkan oleh komputer.

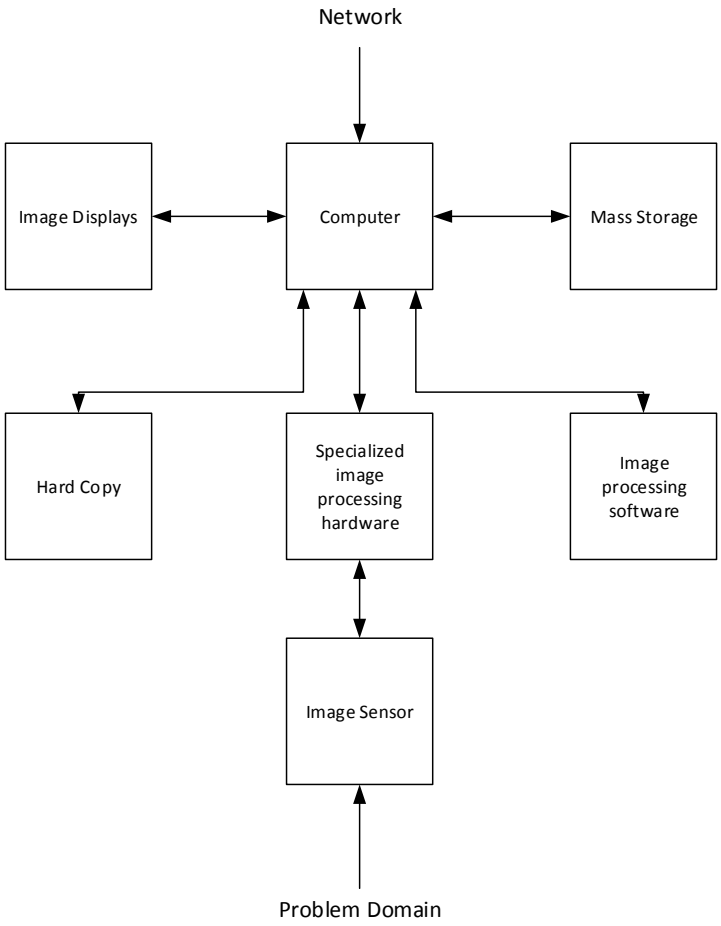
Citra didasarkan pada radiasi spektrum EM yang paling familier, khususnya citra X-ray dan band visual dari spektrum. Gelombang elektromagnetik dapat dikonsepkan sebagai propagasi gelombang sinusoidal dari macam-macam panjang gelombang [6].

2.3.3 Komponen Pengolahan Citra

Pertengahan 1980, banyak model dari sistem pengolahan citra dijual di seluruh dunia dimana peralatan peripheral yang lebih penting dimasukan untuk mengimbangi perangkat komputer yang penting. Akhir tahun 1980 dan awal 1990, pasar digeser ke *hardware* pengolahan citra dalam bentuk *single board*yang didesain kompatibel standar bisnis industri dan dimasukan ke *wordstation* teknisi dan komputer personal (PC). Untuk menekan harga, pasar ini juga menggeser *catalyst* untuk sejumlah komputer perusahaan yang dikhususkan untuk pengembangan software pengolahan citra.

Walaupun sistem pengolahan citra berskala besar untuk aplikasi citra *massive*, seperti pemrosesan citra satelit, tren dilanjutkan dengan *miniaturizing* dan *blending* komputer serbaguna yang kecil dengan perangkat keras pengolahan citra khusus.

Komponen dasar yang terdiri dari *general-purpose system* yang digunakan untuk pengolahan citra digital dapat dilihat pada gambar berikut:



Gambar 2.3 Komponen Sistem Pengolahan Citra Digital Serbaguna

Dua komponen yang dibutuhkan untuk mengambil citra digital adalah :

Perangkat fisik yang sensitif energi yang diradiasi oleh obyek yang diinginkan untuk menjadi citra.

Digitalizer, perangkat untuk mengkonversi keluaran dari perangkat fisik ke dalam bentuk digital.

Specialized image processing hardware biasanya berisi *digitalizer*, ditambah hardware yang mengerjakan operasi dasar lain, seperti *Arithmethic Logic Unit* (ALU) yang melakukan operasi aritmatika dan logika secara paralel pada citra. Satu contoh adalah bagaimana ALU digunakan untuk *averaging* citra secepat citra didigitasi, dengan tujuan pengurangan *noise*. Jenis hardware ini kadang disebut dengan *front-end subsystem*, dan yang paling membedakan karakteristiknya adalah kecepatan. Dengan kata lain, unit ini melakukan fungsi

yang membutuhkan data cepat (seprti *digitizing* dan *averaging* citra video 30 fps) sedangkan komputer biasa tidak dapat melakukannya.

Komputer dalam pengolahan citra adalah *general-purpose computer* dan dapat dijangkau oleh PC sampai *super computer*. Dalam *dedicated application*, komputer yang dikostumisasi digunakan untuk mendapatkan level kinerja yang diinginkan, tetapi yang digunakan disini adalah sistem pengolahan citra yang serbaguna. Dalam sistem ini, cocok juga untuk pekerjaan pengolahan citra *offline*.

Software untuk pengolahan citra terdiri dari modul khusus yang mengerjakan pekerjaan tertentu. Paket yang didesain dengan baik juga termasuk kemampuan bagi user untuk menulis kode utilisasi modul khusus.

Kemampuan penyimpanan masal harus ada dalam aplikasi pengolahan citra. Citra dengna ukuran 1024 x 1024 piksel, dimana intensitas setiap piksel adalah kauntitas 8-bit, membutuhkan satu megabyte tempat penyimpanan jika citra tidak dikompres. Jika mencapai ribuan atatu bahkan juataan citra, penyedia tempat penyimpanan yang cukup menjadi hambatan. Penyimpanan digital untuk aplikasi pengolahan citra terdapat tiga prinsip kategori yaitu:

1. *Shot-term storage* untuk diguanakan selama proses
2. *On-line storage* untuk pengambilan kembali secara relatif cepat
3. *Archival storage* yang dikarakteristikan dengan pengaksesan yang jarang.

Satu metode yang menyediakan *short-term storage* adalah memori, sedangkan yang lain oleh *board* khusus yang disebut dengan *frame buffer,* yang menyimpan satu atau lebih citra dan dapat diakses secara cepat, biasanya pada *video rates* (misalnya 30 citra lengkap per detik). Metode yang terakhir secara virtual membolehkan saat itu juga pembesaran citra, seperti halnya *scroll* (geser vertikal) dan *pan* (geser horisontal). *Frame Buffer* biasanya disimpan dalam unit *hardware* khusus pengolahan citra. *On-line storage* umumnya berbentuk disk magnetik atau *optical-media storage.* Faktor kunci pengkarakteristikan *n-line storage* sering terhadap data ygn disimpan. *Archival storage* dikarakteristikan dengan kebutuhan tempat penyimpanan masif yang jarang diakses. Tape magnetik dan *optical disk house* dalam *jukebox* adalah media yang biasa untuk aplikasi perngarsipan.

Perangkat *hardcopy* untuk merekam citra, termasuk laser printer, kamera film, *heat-sensitive devices, inkjet unit,* dan unit digital seperti disk optikal dan CD-ROM/DVD-ROM. Film memberikan kemungkinan resolusi paling tinggi, sedangkan kertas adalah media yang jelas untuk pilihan penulisan bahan. Untuk presentasi, citra ditampilkan pada film transparan atau media digital jika perangkat proyeksi citra digunakan.

Networking adalah kunci standar pada semua sistem komputer yang digunakan saat ini karena data yang besar tidak dapat dipisahkan dalam aplikasi pengolahan citra. Kunci yang diperhatikan dalam transmisi citra adalah *bandwith*. Dalam *dedicated network*, hal ini tidak menjadi masalah, tetapi komunikasi dengan *remote site* lewat internet tidak selalu efisien. Namun situasi ini meningkat lebih baik dengan adanya serat optik dan teknologi *boarband* lain [6].

2.4 Perbaikan Kualitas Citra

Perbaikan kualitas citra bisa memiliki beberapa tujuan diantaranya :

1. Citra memiliki tampilan yang lebih baik menurut selera manusia.
2. Citra menjadi lebih mudah untuk dianalisis oleh proses otomatisasi berbasis citra.
3. Menghilangkan artifak-artifak pengganggu yang tidak diinginkan atau lebih dikenal dengan istilah derau.

Proses-proses yang termasuk kedalam perbaikan citra (*image enhancemenct*) diantaranya :

1. Pengubahan kecerahan citra (*image brightness*).
2. Perbaikan kontras.
3. Pengurangan derau.
4. Pelembutan citra (*image smoothing*).
5. Penajaman citra (*image sharpening*) [5].

2.4.1 Perbaikan Kontras

Terdapat beberapa teknik yang dapat dilakukan untuk memperbaiki kontras pada citra. Teknik pertama adalah *histogram streching*, yaitu histogram ditarik sedemikian rupa sehingga lebih merata distribusinya. Teknik kedua adalah histogram equalization, yaitu mirip dengan teknik pertama yang bertujuan memuat distribusi pada histogram tersebar lebih merata. Namun yang membedakan adalah pada tekniknya [5].

2.4.2 Pengurangan Derau

Derau termasuk ke dalam bagian citra berfrekuensi tinggi. Penggunaan penghalusan citra yang berjenis *low pass filter* mengakibatkan derau yang berfrekuensi tinggi akan terblok atau minimal terkurangi. Low pass filter akan meloloskan bagian citra yang berfrekuensi rendah seperti bagian objek yang berwarna homogen atau bergradasi, dengan filter ini bagian citra berfrekuensi tinggi sepert tepi citra akan terdegradasi sehingga terlihat lebih lembut, beberapa contoh filter yang bisa digunakan adalah *mean filter* dan *median filter*. Selain untuk melembutkan citra, kedua filter tersebut dapat digunakan untuk mengurangi derau [5].

2.4.3 Jenis-Jenis Derau

Noise atau derau secara sederhana adalah suatu gangguan atau kecacatan pada citra yang mengakibatkan citra tidak jelas dan kehilangan informasi. Penyebab adanya derau pada citra antara lain karena :

1. Proses akusisi yang kurang baik, seperti kurang atau lebih cahaya yang tertangkap
2. Sensor yang digunakan
3. Lingkungan, seprti terdapat awan ketika pengambilan melalui citra satelit
4. Transmisi, terjadi *data loss* ketika pengiriman citra melalui sebuah sinyal yang mengakibatkan citra yang diterima kurang atau tidak sama dengan citra yang dikirim [5].

2.4.4 Salt and Pepper

Derau *salt and pepper* adalah derau yang sangat sering terjadi. Bentuk dari derau ini adalah adanya bintik-bintik hitam dan putih pada citra *grayscale*. Bintik putih untuk *salt* dan bintik hitam untuk *pepper*. Namun namun bila derau ini terdapat pada citra berwarna maka akan berbentuk bintik-bintik berwarna warni [5].

Sumber lain mengatakan derau *salt & pepper* biasa dinamakan sebagai derau impuls positif dan negatif, derau tembakan, atau derau biner. Derau ini biasa disebabkan oleh gangguan yang tiba-tiba dan tajam pada proses perolehan isyarat citra. Derau *salt & pepper* sering muncul pada citra yang diperoleh melalui kamera [7].

2.4.5 Gaussian

Derau gaussian adalah derau yang terdistribusi pada citra dengan distribusi normal. Kepadatan deraunya bergantung pada fungsi kepadatan probabilitas. Penyebab derau ini antara lain pencahayaan yang kurang, suhu yang sangat tinggi, dan kesalahan dalam transmisi citra [5].

2.4.6 Periodik

Derau periodik bersifat periodik (bukan acak). Secara visual derau periodik berbentuk garis-garis pada citra [5].

Sumber lain mengatakan derau periodis biasa terjadi karena interfensi listrik maupun elektromekanis selama citra diakuisisi. Derau ini biasanya berbentuk sinusoidal. Sifat periodis dapat berbentuk statisioner yaitu memiliki amplitudo, frekuensi, dan fase tetap, tetapi dapatjuga non statisioner dengan nilai amplitudo, frekuensi, dan fase berubah di sepanjang area citra [7].

2.5 Median Filter

Filter median sangat populer dalam pengolahan citra. Filter ini dapat dipakai untuk menghilangkanderau bintik-bintik. Nilai yang lebih baik digunakan untuk suatu piksel ditentukan oleh nilai median dari setiap piksel dan kedelapan piksel tetangga pada 8-ketetanggaan [7].

Secara matematis, filter dapat dinotasikan sebagai berikut [7]:

$$g(y,x)=median\left(f(y-1,x-1),\ f(y-1,x),\ f(y-1,x+1),\ f(y,x-1),\ f(y,x),\ f(y,x+1),\ f(y+1,x-1),\ f(y+1,x),\ f(y+1,x+1) \right) \quad (2.1)$$

Median filter beroperasi dengan cara mengintegrasikan sebuah jendela dengan ukuran tertentu dari pojok kiri atas sampai pojok kanan bawah citra. Operasi yang dilakukan yaitu pada setiap iterasi nilai tengah (median) dari nilai piksel-piksel citra yang berada di dalam jendela tersebut. Ukuran jendela bisa 3x3, 5x5 dan seterusnya dengan syarat berukuran ganjil [5].

Berikut adalah penjelasan dari cara kerja *Median Filter* dapat dilihat sebagai berikut:

186	184	197	188
160	78	196	112
171	181	159	178
171	183	191	120

Gambar 2.4 Nilai Piksel Citra

Contoh citra diatas memiliki nilai-nilai piksel, dan menggunakan 3x3 *mask filtering*. Langkah selanjutnya adalah mengurutkan nilai piksel yang berada dalam *mask filter* mulai dari nilai piksel terkecil hingga ke nilai piksel terbesar.

Bila diurutkan maka akan menjadi berikut :

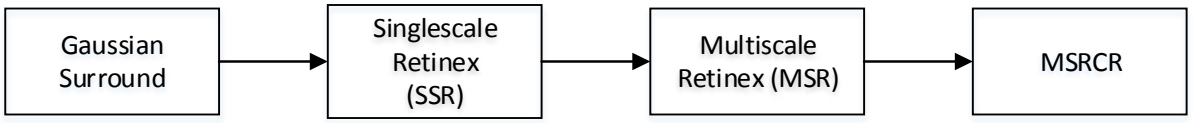
78, 159, 160, 171, 181, 184, 186, 196, 197

Setelah diurutkan maka diambil nilai piksel yang berada di posisi tengah yaitu 181 dan bertukar posisi dengan nilai piksel 78 pada pusat *mask filtering*.

2.6 Multiscale Retinex with Color Restoration (MSRCR)

Retinex teori dijelaskan oleh Edwin H. Land pada tahun 1971. Kata "*Retinex*" adalah penggabungan kata yang terbentuk dari kata "*Retina*" dan "*Cortex*", yang menunjukan bahwa baik mata dan otak terlibat dalam pemrosesan. Teori Retinex yaitu teori yang berdasarkan ketetapan warna "*Color Constancy*" yang didasarkan bahwa persepsi mata manusia pada warna tergantung pada karakteristik refleksi permukaan objek, dan tidak berhubungan dengan kejadian tersebut.

Terdapat beberapa perkembangan pada teori *Retinex* yaitu, *Single-Scale Retinex* (SSR), *Multi-Scale Retinex* (MSR), dan *Multi-Scale Retinex with Color Restoration* (MSRCR). MSRCR memiliki hasil lebih baik dari tipe *Retinex* yang lain, namun tidak dapat mempertahankan ketetapan corak sehingga masih ada sedikit distorsi warna [8]. Adapun tahapan yang dilakukan pada MSRCR adalah sebagai berikut :



Gambar 2.5 Tahapan MSCRCR

2.6.1 Gaussian Surround

Gaussian Surround digunakan sebagai *filter* untuk memperhalus citra asli. Tahap ini akan digunakan untuk proses SSR [9]. Adapun persamaan dari Gaussian Surround adalah sebagai berikut:

$$F(x,y)=Ke^{-(x^2+y^2)/(2\times\sigma^2)} \quad (2.2)$$

Berikut penjelasan dari persamaan tersebut:

F(x,y) = Gaussian pada piksel (x,y)

e = eksponensial

(x,y) = koordinat piksel

σ = nilai sigma

2.6.2 Singlescale Retinex (SSR)

Multiscale Retinex (MSR) pada dasarnya menggunakan metode Singlescale Retinex (SSR) namun dengan skala yang berbeda [9]. Adapun persamaan dari SSR adalah sebagai berikut:

$$R_i(x,y) = \log I_i(x,y) - \log[F(x,y) * I_i(x,y)] \quad \text{.....(2.3)}$$

Berikut penjelasan dari persamaan tersebut :

$R_i(x,y)$ = *output Retinex*

$I_i(x,y)$ = distribusi citra pada piksel ke i(x,y)

$*$ = operator konvolusi

i = kanal warna (R,G,B)

2.6.3 Multiscale Retinex (MSR)

Ide dari dasar Multi-Scale Retinex untuk masalah SSR terdapat pada skala yang berbeda yang digunakan dan diberi bobot yang berbeda untuk menggabungkan kelebihan dan menghilangkan kelemahan yang dimiliki dari skala rendah dan skala besar [9]. Adapun persamaan dari MSR adalah sebagai berikut:

$$R_{MSR_i}(x,y) = \sum_{n=1}^{N} W_n R_{n_i} \quad \text{.....(2.4)}$$

Berikut penjelasan dari persamaan tersebut :

R_{MSR_i} = output dari MSR

N = jumlah skala

Wn = bobot yang diasosiasikan dengan skala ke-n

R_{n_i} = output dari SSR yang diasosiasikan dengan skala ke-n

i = kanal warna (R,G,B)

2.6.4 Multiscale Retinex with Color Restoration (MSRCR)

MSRCR merupakan pengembangan dari MSR, dimana metode ini mampu memperbaiki kualitas citra yaitu pada pencerahan citra dengan mempertahankan *color constancy*. *Color constancy* atau ketetapan diambil dari sistem penglihatan

manusia yang mengusahakan warna dari suatu benda teteap terlihat sama walaupun pada kondisi pencahayaan yang berbeda-beda [9]. Adapun persamaan dari MSRCR adalah sebagai berikut:

$$C_i(x,y) = \beta\{\log[\alpha\, I_i(x,y)] - \log[\sum_{i=1}^{S} I_i(x,y)]\} \quad (2.5)$$

$$R_{MSRCR_i}(x,y) = G\left[C_i(x,y)R_{MSR_i}(x,y) + b\right] \quad (2.6)$$

Berikut penjelasan dari persamaan tersebut:

β = konstanta *gain*

α = konstanta kontrol ketidaklinieran

S = 3, kanal warna (RGB)

G dan b = nilai akhir *gain* dan *offset*

2.7 Object Oriented Programming

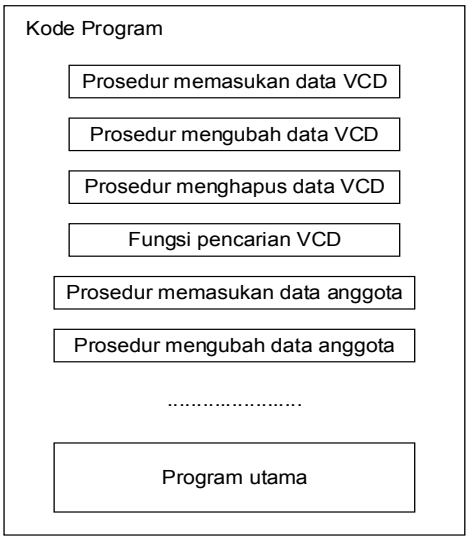
Object Oriented Programming atau pemrograman berorientasi objek adalah sebuah konsep pemrograman untuk membuat kode program yang lebih terstruktur, terkelompok berdasarkan objek-objek yang terlibat sehingga bagian-bagiannya dapat digunakna untuk pembuatan aplikasi lain. Pemrograman berorientasi objek membagi-bagi kode program aplikasi menjadi kumpulan objek dipandang dari sudut pandang aplikasi komputer dan proses yang dilakukan didalam aplikasi [9].

2.7.1 Pemrograman Prosedural Dan Pemrograman Berorientasi Objek

Pemrograman prosedural merupakan pemrograman yang membagi-bagi proses yang dilakukan program menjadi prosedur-prosedur atau fungsi-fungsi yang dibuat berdasarkan pembagian proses yang dibutuhkan sebuah program.

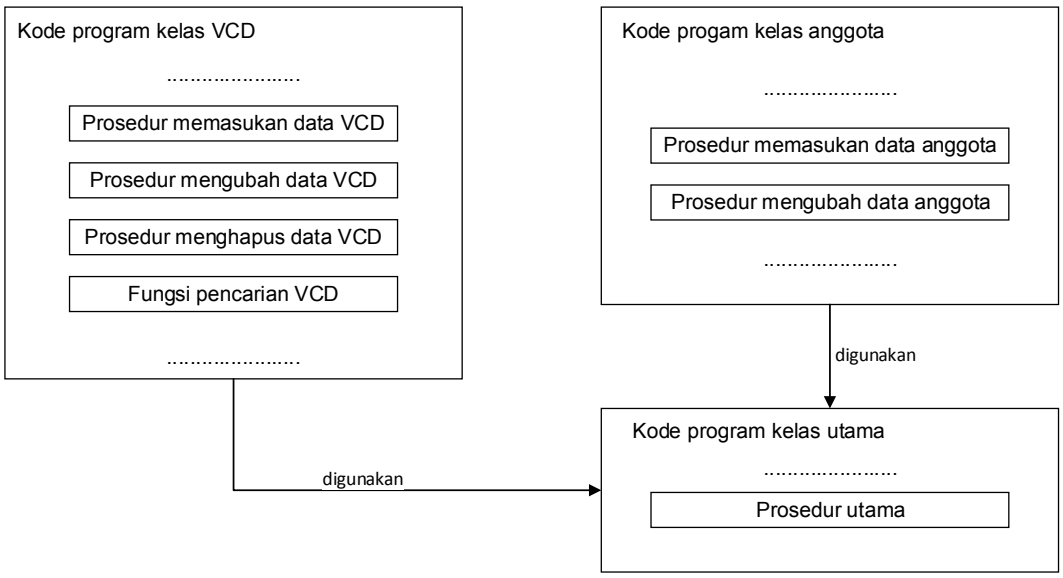
Sedangkan pemrograman berorientasi objek membagi-bagi kode program menjadi kelas-kelas yang dibuat berdasarkan objek-objek yang terlibat dalam pengembangan program [9].

Berikut adalah gambaran pemrograman prosedural dan pemrograman berorientasi objek pada contoh kasus penyewaan VCD. Gambaran blok kode program prosedural dapat dilihat pada gambar 2.6



Gambar 2.6 Blok Kode Program Prosedural

Gambaran blok kode program berorientasi objek dapat dilihat pada gamabar 2.7



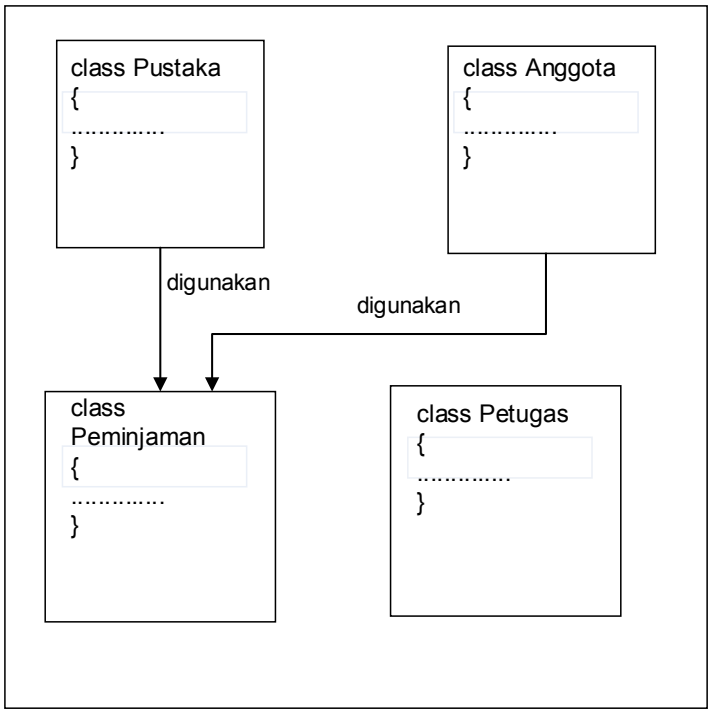
Gambar 2.7 Blok Kode Program Berorientasi Objek

2.7.2 Kelas

Kelas adalah kumpulan objek-objek dengan karakterisik yang sama. Kelas merupakan definisi statik dan himpunan objek yan gsama yang mungkin lahir atau diciptakan dan kelas tersebut. Sebuah kelas akan mempunyai sifat (atribut), kelakuan (operasi/metode), hubungan (relationship) dan arti. Suatu kelas dapat

diturunkan dan kelas yang lain, dimana atribut dan kelas semula dapat diwariskan ke kelas yang baru [9].

Secara teknis, kelas adalah sebuah struktur tertentu dalam pembuatan perangkat lunak. Kelas merupakan bentuk struktur pada kode program yang menggunakan metodologi berorientasi objek. Kelas secar fisik adalah berkas yang berisi kode program, dimana kode program merupakan semua hal yang terkait dengna nama kelas. Ilustrasi dari sebuah kelas adapat dilhat pada gambar 2.8



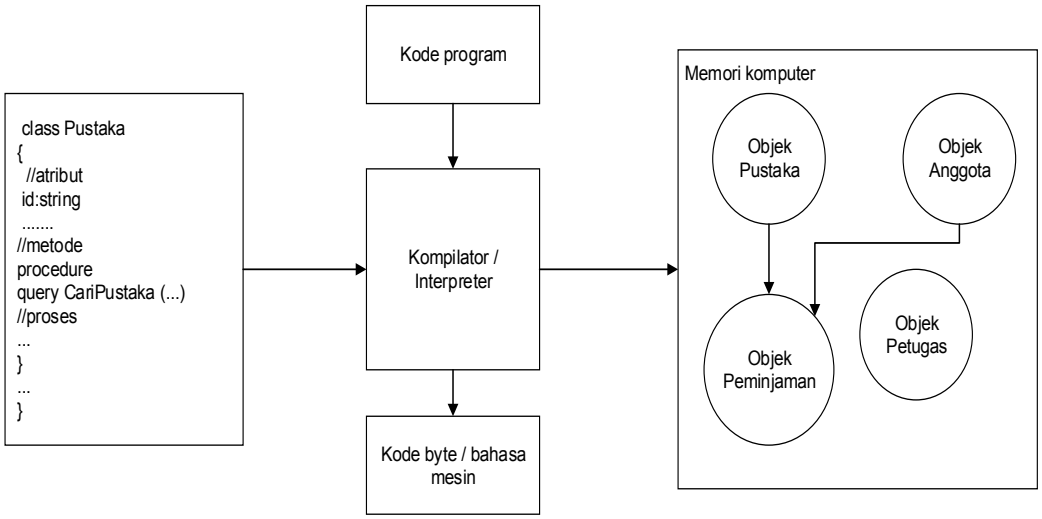
Gambar 2.8 Ilustrasi Kelas

2.7.3 Objek

Objek adalah abstraksi dan sesuatu yang mewakili dunia nyata seperti benda, manusia, satuan organisasi, tempat, kejadian, struktur, status, atau hal-hal lain yang bersifat abstrak. Objek merupakan suatu entitas yang mampu menyimpan informasi (status) dan mempunyai operasi (kelakuan) yang dapat diterapkan atau dapat berpengaruh pada status objeknya. Objek mempunyai siklus hidup yaitu diciptakan, dimanipulasi, dan dihancurkan.

Secara teknis, sebuah kelas saat program dieksekusi maka akan dibuat sebuah objek. Objek dilihat dari segi teknis adalah elemen pada saat *runtime* yang

akan diciptakan, dimanipulasi dan dihancurkan saat eksekusi sehingga sebuah objek hanya ada saat sebuah program dieksekusi, jika masih dalam bentuk kode, disebut sebagai kelas jadi pada saat *runtime*. [9] Ilustrasi keterkaitan kelas dan objek dapat dilihat pada gambar 2.9



Gambar 2.9 Ilustrasi Kelas Dan Objek

2.7.4 Metode

Operasi atau metode pada sebuah kelas hampir sama dengan fungsi atau prosedur pada metodologi struktural. Sebuha kelas boleh memiliki lebih dari satu metode atau operasi. Metode atau operasi yang berfungsi untuk memanipulasi objek itu sendiri. Operasi atau metode merupakan fungsi atau transformasi yang dapat dilakukan terhadap objek atau dilakukan oleh objek.

Metode atau operasi dapat berasal dari *event*, aktifitas atau aksi keadaan, fungsi, atau kelakuan dunia nyata. Contoh metode atau operasi misalnya Read, Write, Move, Copy dan sebagainya. Kelas sebaiknya memiliki metode get dan set unutk setiap atribut agar konsep enkapsulasi tetap terjaga.

Metode get digunakan untuk memberikan akses lain dalam mengakses atribut, dan set adalah metode yang digunakan untuk mengisi, agar kelas lain tidak mengakses atribut secara langsung [9].

2.7.5 Atribut

Atribut dari sebuah kelas adalah variabel global yang dimilik sebuah kelas. Atribut dapat berupa nilai atau elemen-elemen data yang dimilliki oleh objek dalam kelas objek. Atribut dipunyai secara individual oleh sebuah objek, misalnya berat, jenis, nama dan sebagainya. Atribut sebaiknya privat untuk menjaga konsep enkapsulasi [9].

2.7.6 Enkapsulasi

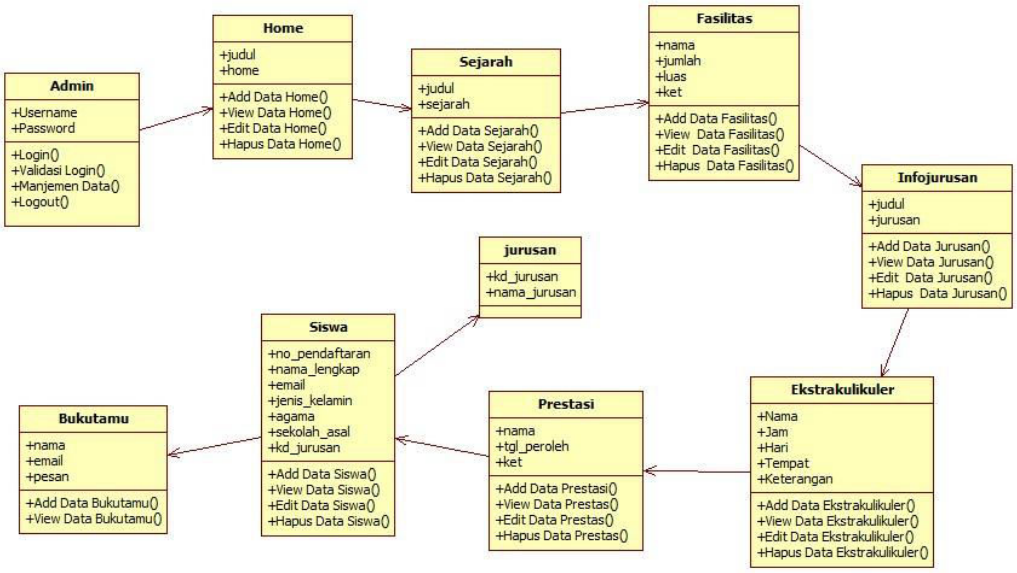
Enkapsulasi adalah pembungkusan atribut data layanan (operasi-operasi) yang mempunyai objek untuk menyembunyikan implementasi dan objek sehingga objek lain tidak mengetahui cara kerjanya [9].

2.8 UML (*Unified Modeling Language*)

UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengna menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML lebih banyak digunakan pada metodologi berorientasi objek [9].

2.8.1 *Class Diagram*

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibat untuk membangun sistem. Class diagram dibuat agar pembuat program membuat kelas-kelas sesuai rancangan didalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron [9]. Contoh kelas diagram dapat dilihat pada gambar 2.10



Gambar 2.10 Contoh Class Diagram

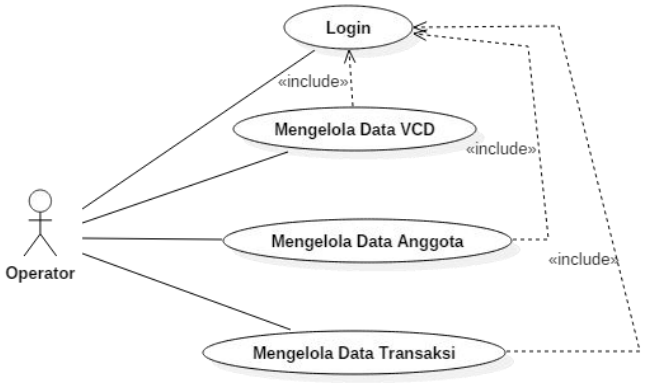
2.8.2 *Use Case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. Usecase mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case.

Aktor merupakan orang, proses, atau sisitem lain yang berinteraksi dengan sistem informasi yang akan dibuat. Namun walaupun aktor disimbolkan dengan gambar orang belum tentu aktor merupakan orang.

Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor [9].

Bentuk use case dapat dilihat pada gambar 2.11

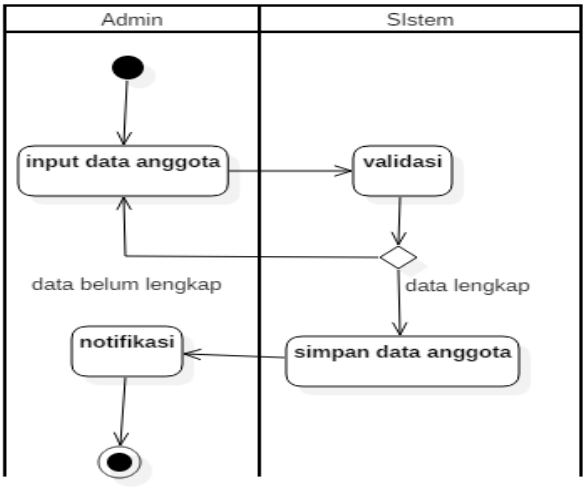


Gambar 2.11 Contoh Use Case Diagram

2.8.3 *Activity Diagram*

Diagram aktifitas atau *activity diagram* menggambarkan aliran kerja atau aktifitas dari sebuah sistem atau proses bisnis atau menu yan gada pada perangkat lunak. Perlu diperhatikan bahwa diagram aktifitas menggmabarkan aktifitas sistem bukan apa yang dilakukan aktor [9].

Contoh aktifitas diagram dapat dilihat pada gambar 2.12



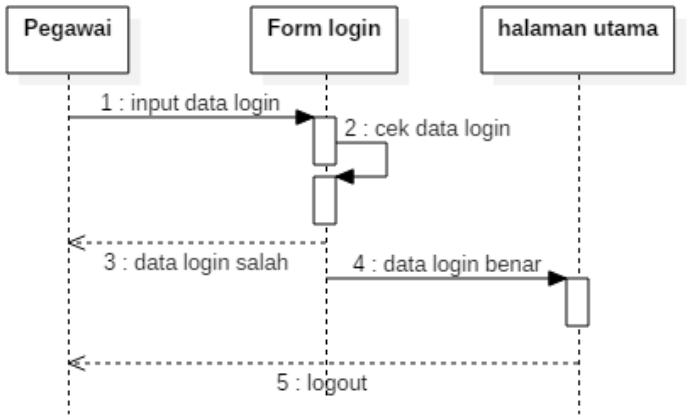
Gambar 2.12 Contoh Activity Diagram

2.8.4 *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak

pendefinisian use case yang memiliki proses sendiri atau yang penting semua use case yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak use case yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak [9].

Contoh *sequence* diagram dapat dilihat pada gambar 2.13



Gambar 2.13 Contoh Sequence Diagram

2.9 MATLAB

MATLAB (Matrix Laboratory) adalah sebuah pemrograman untuk analisis dan komputasi numerik, merupakan suatu bahasa pemrograman matematika lanjutan yang dibentuk dengan dasar pemikiran menggunakan sifat dan bentuk matriks. Pada awalnya, program ini merupakan interface untuk koleksi rutin-rutin numerik proyek LINPACK dna EISPACK, dikembangkan menggunakan bahasa FORTRAN. Namun sekarang, program ini merupakan produk komersial dari perusahaan Mathworks, Inc. yang dalam perkembangan selanjutnya dikembangkan menggunakan bahasa C++ dan assembler (terutama untuk fungsi-fungsi dasar MATLAB).

Matlab telah berkembang menjadi sebuah environment pemrograman yang canggih dan berisi fungsi-fungsi built-in untuk melakukan tugas pengolahan sinyal, aljabar linear, dan kalkulasi matematis lainnya. Matlab juga berisi toolbox yang berisi fungsi-fungsi tambahan untuk aplikasi khusus. Matlab bersifat extensible, dalam artibahwa seorang pengguna dapat menulisfungsi baru untuk

ditambahkan di library jika fungsi-fungsi built-in yang tersedia tidak dapat melakukan tugas tertentu. Kemampuan pemrograman yang dibutuhkan tidak terlalu sulit bila anda telah memiliki pengalaman dalam pemrograman bahasa lain seperti C, Pascal atau fortran.

Matlab merupakan software yang dikembangkan oleh Mathworks, Inc. dan merupakan software yang paling efisien untuk perhitungan numerik berbasis matriks. Dengan demikian jika didalam perhitungan kita dapat memformulasikan masalah kedalam format matriks, maka matlab merupakan software terbaik untuk penyelesaian numeriknya.

Matlab yang merupakan bahasa pemrograman tingkat tinggi berbasis pada matriks sering digunakan untuk teknik komputasi numerik, digunakan untuk menyelesaikan masalah-masalah yang melibatkan operasi matematika elemen, matriks, optimasi, aproksimasi dan lain-lain. Matlab banyak digunakan pada :

1. Matematika dan Komputasi
2. Pengembangan dan Algoritma
3. Pemrograman *Modelling,* simulasi, dan pembuatan prototipe
4. Analisis Data, eksplorasi, dan visualisasi
5. Analisis numerik dan statistik
6. Pengembangna aplikasi teknik [10]

Karakteristik Matlab antara lain :

1. Bahasa pemrogramannya didasarkan pada matriks (baris dan kolom).
2. Lambat (dibandingkan dengan Fortran atau C) karena bahasanya langsung diartikan. Sebagai contoh, tidak diperlukan pre-compiled. Menghindari kalang for (for loops). Setiap saat menggunakan bentuk-bentuk vektor.
3. Automatic memory management,misalnya kita tidak harus mendeklarasikan arrays terlebih dahulu.
4. Tersusun rapi (seperti pegaturan array di Fortran-90)
5. Memiliki waktu pengembangan program yang lebih cepat dibandingkan bahasa pemrograman tradisional seperti Fortran atau C.
6. Dapat diubah ke bahasa C lewat matlab compiler untuk efisiensi yang lebih baik.

7. Tersedia banyak toolbox untuk aplikasi-aplikasi khusus.
8. Bersama dengan Maple untuk komputasi-komputasi simbolik.
9. Dalam shared-memory parallel computers, seperti SGI Origin2000, beberapa operasi secara otomatis dapat diproses bersama [11].

2.9.1 Sejarah MATLAB

Cleve Moler adalah pencipta MATLAB. Moler adalah seorang analis numerik yang mengkhususkan diri dalam perhitungan matriks. Pada awal tahun 1970, Moler berpartisipasi dalam upaya yang diselenggarakan oleh Argonne National Laboratory untuk menghasilkan perpustakaan subrutin FORTRAN yang andal dan efisien untuk menghitung nilai eigen matrik (EISPACK) dan sistem penyelesaian persamaan linier (LINPACK). Agar paket ini lebih mudah digunakan oleh siswa, Moler menciptakan MATLAB, yang namanya merupakan kontraksi dari "Matrix Laboratory." Sebenarnya MATLAB adalah program FORTRAN, dirancang untuk era terminal timesharing dan ASCII. MATLAB hanya memiliki satu tipe data (matriks ganda kompleks) dan kumpulan tetap dari 80 fungsi. Moler menggunakan FORTRAN MATLAB dalam mengajarkan analisis numerik di Stanford pada tahun 1979. Jack Little mengetahui kabar dari program tersebut, yang memahami potensi MATLAB dalam pemrosesan dan pengendalian sinyal, dan kemungkinan membuat produk perangkat lunak yang sukses berdasarkan MATLAB untuk PC IBM yang baru.

Little dan Steve Bangert mengembangkan PC MATLAB dengan memetakan kode Moler dari FORTRAN ke C, menambahkan fungsi yang ditentukan pengguna (yang pada dasarnya membuat MATLAB menjadi bahasa pemrograman daripada kalkulator), grafik yang lebih baik, *libraries of MATLAB routines*, *toolboxes*. Moler, Little, dan Bangert membentuk Mathworks pada tahun 1984, dengan PC MATLAB sebagai produk pertama. Penjualan pertama mereka adalah pesanan untuk sepuluh eksemplar yang ditempatkan oleh Profesor Nick Trefethen di MIT.