BAB II

LANDASAN TEORI

Teori merupakan dasar yang digunakan dalam pembangunan perangkat lunak. Pada bab ini akan dijelaskan mengenai landasan teori yang berkaitan erat dengan perangkat lunak yang dibangun.

2.1. Multimedia

Komputer sebagai alat pengolahan *digital* saat ini hampir dimiliki oleh setiap orang, seperti halnya televisi, komputer juga banyak digunakan sebagai media hiburan. Dengan perkembangan komputer (serta pengetahuan dalam pengolahan sinyal *digital*), maka data-data dalam bentuk *digital* semakin banyak digunakan.

Penggunaan data *digital* baik berupa text, suara, citra maupun video sangat pesat, ditambah dengan perkembangan teknologi jaringan antar komputer didunia (internet), pertukaran data *digital* semakin mudah untuk dapat menggabungkan semua data *digital* yang ada (multimedia).

Dalam pembuatan aplikasi-aplikasi multimedia diperlukan beberapa komponen pendukung seperti *text,* suara / audio, dan citra */ image.*

2.1.1. Data *Text*

Text merupakan sekumpulan karakter terdiri dari huruf-huruf, angka-angka (A-Z, a-z, 0-9), dan simbol-simbol lainnya seperti %, &, ^, =, @, £, $, !,*, dan lain-lain, dengan menggunakan kode ASCII setiap karakter dari *text* berjumlah 8 bit atau 1 *byte.*

2.1.2. Data *Image*

Citra / *images* adalah gambar *digital* dua dimensi yang dihasilkan dari gambar analog, dibagi kedalam N baris dan M kolom sehingga menjadi gambar diskrit melalui proses *sampling* (proses pengambilan *sample* warna sebelum melakukan proses digitalisasi citra), dan persilangan antara baris dan kolom disebut dengan *pixel. Images* diklasifikasikan ke dalam dua jenis yaitu :

1. *Images* Tetap

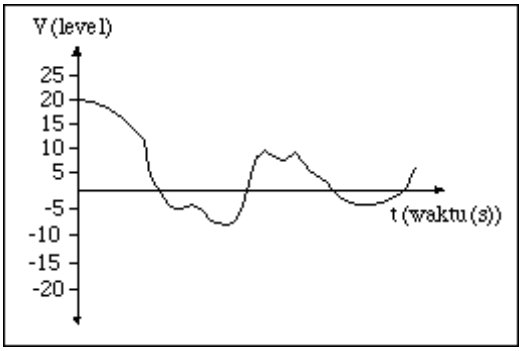
   Images / gambar mati (grafik statis) yang tidak mengalami proses animasi biasanya berupa format BMP, JPG, JPEG, ICO dan lain-lain.

2. *Images* Bergerak /Animasi

   Image bergerak adalah *image* tetap yang diberikan efek seolah-olah seperti bergerak, atau sering disebut animasi. Animasi adalah proses kejadian dalam pembuatan film (*movie*), saat ini banyak film animasi tiga dimensi daripada dua dimensi.

2.1.3. Data Suara

Suara (audio) atau bunyi merupakan sebuah sinyal analog yang dihasilkan oleh getaran yang membentuk sebuah gelombang *longitudinal* dengan frekuensi 20 hz sampai 20.000 hz (berdasarkan tingkat pendengaran manusia).



Gambar 2.1. Suara dalam bentuk sinyal analog

(Sumber : Drs Sunomo [9])

Gelombang suara yang masih berbentuk sinyal analog dapat diubah kedalam bit-bit *digital* menjadi sinyal *digital*, melalui proses *sampling* dan digitalisasi.

2.2. Steganografi

Steganografi (*steganography*) adalah ilmu dan seni menyembunyikan pesan rahasia (*hiding message*) sedemikian sehingga keberadaan (eksistensi) pesan tidak terdeteksi oleh indera manusia. Kata steganorafi berasal dari Bahaya Yunani yang berarti “tulisan tersembunyi” (*covered writing*). Steganografi membutuhkan dua properti: wadah penampung dan data rahasia yang akan disembunyikan. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, suara, teks, dan video. Data rahasia yang disembunyikan juga dapat berupa citra, suara, teks, atau video. Steganografi berbeda dengan

kriptografi, di mana pihak ketiga dapat mendeteksi adanya data (*chipertext*), karena hasil dari kriptografi berupa data yang berbeda dari bentuk aslinya dan biasanya datanya seolah-olah berantakan, tetapi dapat dikembalikan ke bentuk semula. Ilustrasi mengenai perbedaan kriptografi dan steganografi dapat dilihat pada gambar 2.2.

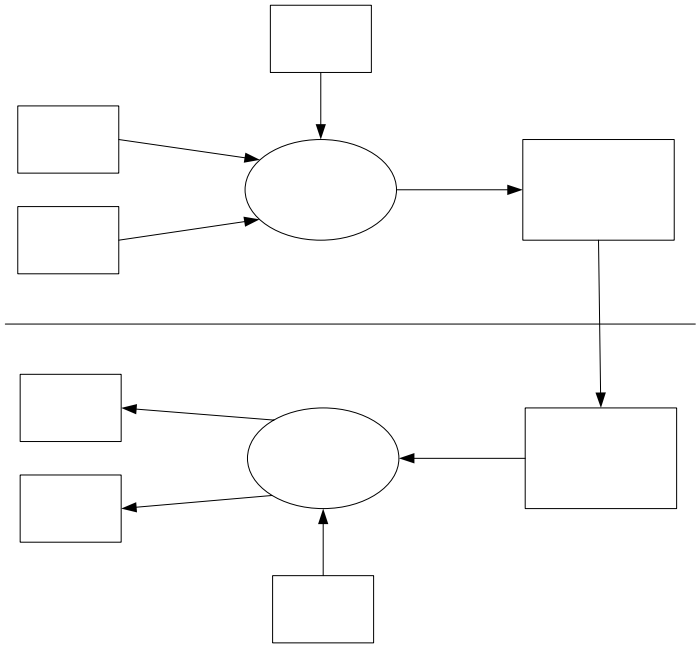


Gambar 2.2. Ilustrasi kriptografi dan steganografi pada citra digital.

Steganografi membahas bagaimana sebuah pesan dapat disisipkan ke dalam sebuah berkas media sehingga pihak ketiga tidak menyadarinya. Steganografi memanfaatkan keterbatasan sistem indera manusia seperti mata dan telinga. Dengan adanya keterbatasan inilah, metoda steganografi ini dapat diterapkan pada berbagai media digital. Hasil keluaran dari steganografi ini memiliki bentuk persepsi yang sama dengan bentuk aslinya, tentunya persepsi di sini sebatas oleh kemampuan indera manusia, tetapi tidak oleh komputer atau perangkat pengolah digital lainnya. Ilustrasi mengenai proses steganografi dapat dilihat pada gambar 2.3.

Penyembunyian data rahasia ke dalam media digital mengubah kualitas media tersebut. Kriteria yang harus diperhatikan dalam penyembunyian data diantaranya adalah :

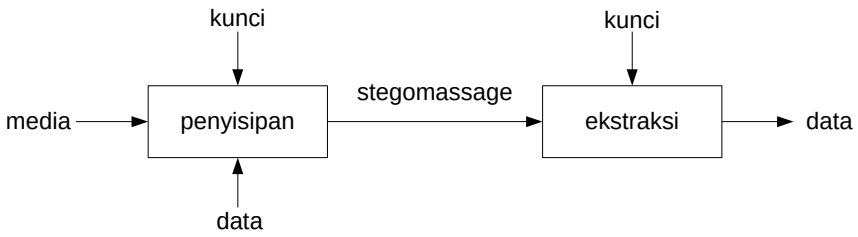
1. *Fidelity*. Mutu citra penampung tidak jauh berubah. Setelah penambahan data rahasia, citra hasil steganografi masih terlihat dengan baik. Pengamat tidak mengetahui kalau di dalam citra tersebut terdapat data rahasia.
2. *Recovery*. Data yang disembunyikan harus dapat diungkapkan kembali (*recovery*). Karena tujuan steganografi adalah *data hiding*, maka sewaktu-waktu data rahasia di dalam citra penampung harus dapat diambil kembali untuk digunakan lebih lanjut.



Gambar 2.3. Diagram Sistem Steganografi

Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, suara, teks, dan video. Sedangkan data rahasia yang disembunyikan dapat berupa berkas apapun. Media yang telah disisipi data disebut stegomessage. Proses penyembunyian data ke dalam media disebut penyisipan (*embedding*), sedangkan proses sebaliknya disebut ekstraksi. Proses

tersebut dapat dilihat pada gambar 2.4. Penambahan kunci yang bersifat opsional dimaksudkan untuk lebih meningkatkan keamanan.



Gambar 2.4. Proses penyisipan dan ekstraksi dalam steganografi

2.2.1. Sejarah Steganografi

Steganografi berasal dari bahasa Yunani yaitu Steganós yang berarti menyembunyikan dan Graptos yang artinya tulisan sehingga secara keseluruhan artinya adalah tulisan yang disebunyikan. Secara umum steganografi merupakan seni atau ilmu yang digunakan untuk menyembunyikan pesan rahasia dengan segala cara sehingga selain orang yang dituju, orang lain tidak akan menyadari keberadaan dari pesan rahasia tersebut.

Steganografi sudah digunakan sejak dahulu kala sekitar 2500 tahun yang lalu untuk kepentingan politik, militer, diplomatik, serta untuk kpntingan pribadi sebagai alat. Beberapa contoh penggunaan steganografi pada masa lampau:

1. Pada tahun 480 sebelum masehi, seseorang berkebangsaan Yunani yaitu Demaratus mengirimkan pesan kepada polis Sparta yang berisi peringatan mengenai penyerangan Xerxes yang ditunda. Teknik yang digunakan adalah dengan menggunakan meja yang telah diukir kemudian diberi lapisan lilin

untuk menutupi pesan tersebut, dengan begitu pesan dalam meja dapat disampaikan tanpa menimbulakn kecurigaan oleh para penjaga.

2. Pada abad ke 5 sebelum masehi, Histaiacus mengirimkan pesan kepada Aristagoras Miletus untuk memberontak terhadap raja Persia. Pesan disampaikan dengan cara mencukur kepala pembawa pesan dan mentato kepalanya dengan pesan tersebut. Kemudian saat rambutnya tumbuh kembali, pembawa pesan dikirimkan dan pada tempat tujuan rambutnya kembali digunduli dan pesan akan terbaca.
3. Penggunaan tinta yang tidak terlihat pada pesan lainnya.
4. Pada perang dunia II, Jerman menggunakan *microdots* untuk berkomunikasi. Penggnaan teknik ini biasa digunakan pada *microfilm* chip yang harus diperbesar sekitar 200 kali.
5. Pada perang dunia II, Amerika Serikat menggunakan suku Indian Navajo sebagai media untuk berkomunikasi.

2.2.2. Manfaat Steganografi

Steganography adalah sebuah pisau bermata dua, ia bisa digunakan untuk alasan-alasan yang baik, tetapi bisa juga digunakan sebagai sarana kejahatan. Steganography juga dapat digunakan sebagai salah satu metode watermarking pada image untuk proteksi hak cipta, seperti juga digital watermarking (fingerprinting). Steganography juga dapat digunakan sebagai pengganti hash. Dan yang terutama, seperti disebutkan sebelumnya, steganography dapat digunakan untuk menyembunyikan informasi rahasia, untuk melindunginya dari

pencurian dan dari orang yang tidak berhak untuk mengetahuinya. Sayangnya, steganography juga dapat digunakan untuk mencuri data yang disembunyikan pada data lain sehingga dapat dikirim ke pihak lain, yang tidak berhak, tanpa ada yang curiga. Steganography juga dapat digunakan oleh para teroris untuk saling berkomunikasi satu dengan yang lain. Sehubungan dengan keamanan sistem informasi, steganography hanya merupakan salah satu dari banyak cara yang dapat dilakukan untuk menyembunyikan pesan rahasia. Steganography lebih cocok digunakan bersamaan dengan metode lain tersebut untuk menciptakan keamanan yang berlapis. Sebagai contoh steganography dapat digunakan bersama dengan enkripsi. Windows dan Unix juga menggunakan steganography dalam mengimplementasikan hidden directory.

2.3. Citra Digital

Citra merupakan fungsi intensitas dalam bidang dua dimensi. Intensitas yang dimaksud berasal dari sumber cahaya. Pada hakekatnya citra yang dilihat oleh mata manusia terdiri atas berkas-berkas cahaya yang dipantulkan oleh benda-benda di sekitar kita.

Suatu citra digital adalah suatu gambar kontinu yang diubah dalam bentuk diskrit, baik koordinat maupun intensitas cahayanya. Kita dapat menganggap suatu citra digital sebagai suatu matriks dimana indeks baris dan kolomnya menyatakan koordinat sebuah titik pada citra tersebut dan masing-masing elemennya menyatakan intensitas cahaya pada titik tersebut. Suatu titik

pada sebuah citra digital sering disebut sebagai “image-element” (elemen citra) “picture-element” (elemen gambar) ataupun “*pixel*”.

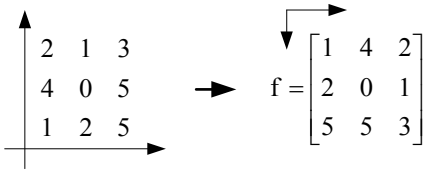
Untuk mengubah suatu citra kontinu ke dalam suatu representasi numerik dilakukan dengan proses digitalisasi oleh suatu *digitizer*, misalnya *scanner*, sehingga citra ini dapat diproses oleh sebuah komputer.

Digitalisasi sebuah citra dilakukan baik terhadap ruang (koordinat (x,y)), maupun terhadap skala keabuannya (f(x,y)). Proses digitalisasi koordinat (x,y) dikenal sebagai “pencuplikan citra” *(image sampling)*, sedangkan proses digitalisasi skala keabuan f(x,y) disebut sebagai “kuantisasi derajat keabuan” *(grey-level quantization)*.

Sebuah citra kontinu f(x,y) akan didekati oleh cuplikan-cuplikan yang seragam jaraknya dalam bentuk matriks MxN, M adalah baris dan N adalah kolom. Nilai elemen-elemen matriks menyatakan derajat keabuan citra, sebangkan posisi elemen tersebut (dalam baris dan kolom) menyatakan koordinat titik-titik (x,y) dari citra. Bentuk matriks di bawah ini dikenal sebagai suatu citra digital.

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ . & . & \dots & . \\ . & . & \dots & . \\ . & . & \dots & . \\ f(M-1,0) & . & \dots & f(M-1,N-1) \end{bmatrix}$$

Matriks di atas dapat disajikan dalam bentuk 2 dimensi dalam sistem koordinat Cartesius dengan memutar posisi matriks di atas sejauh 90° derajat searah jarum jam.



Sedangkan derajat keabuan [0,L] dibagi kedalam G selang dengan panjang selang yang sama, yaitu: $\mathbf{G = 2^m}$ dimana m adalah kedalaman *bit* dan m bilangan bulat positif, bila hal ini diterapkan pada penyimpanan maka sebuah citra digital membutuhkan sejumlah b *bit*, dengan :

$$\mathbf{b = M \times N \times m.}$$

Dalam suatu proses pencuplikan dan kuantisasi sering terjadi permasalahan, yaitu jumlah cuplikan dan derajat keabuan yang diperlukan untuk memperoleh suatu citra yang "baik", makin tinggi nilai MxN dan m, maka citra kontinu f(x,y) akan makin didekati oleh citra digital yang dihasilkan. Tapi hal ini seringkali dibatasi oleh kemampuan hardware dari suatu komputer.

2.3.1. Bitmap (BMP)

BMP atau *DIB* (*device independent bitmap*), adalah sebuah format grafik yang digunakan secara internal oleh microsoft windows dan subsistem OS/2, dan sering digunakan sebagai format file grafik sederhana pada *flatform-flatform* tersebut.

Gambar secara umum disajikan dalam ketajaman warna 2 (1-*bit*), 16 (4-*bit*), 256 (8-*bit*), 65.536 (16-*bit*), 16.7 juta (24-*bit*) warna (*bit-bit* ini mempresentasi *bit-bit* per-*pixel*). Sebuah gambar 8-*bit* juga dapat dirubah ke warna *grayscale* disamping warna indeks. Sebua *channel alpha* (untuk warna

transparan) boleh disajikan dalam *file* terpisah, dimana sama dengan gambar *grayscale*. Versi 32-*bit* dengan *channel alpha* terintegrasi telah diperkenalkan oleh Windows XP dan digunakan untuk sistem *login* dan *theme*.

Umumnya file *BMP* menggunakan model warna *RGB*. Pada model ini sebuah warna terbentuk dari campuran intensitas yang berbeda (bervariasi dari 0 sampai 255), warna merah (R), hijau (G), dan biru (B). Dengan kata lain sebuah warna akan didefinisikan menggunakan 3 nilai, yaitu R, G dan B. Blok dari *bit* mendeskripsikan gambar secara *pixel* per *pixel*. *Pixel* disajikan mulai dari sudut kiri bawah berjalan dari kiri ke kanan dan kemudian baris per baris dari bawah ke atas. Setiap *pixel* dideskripsikan menggunakan satu atau lebih *bit*.

2.4. Video

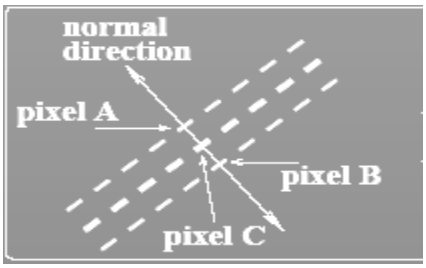
Video merupakan sebuah film atau gambar hidup yang dihasilkan dengan rekaman dari orang dan benda (termasuk fantasi dan *figure* palsu) dengan menggunakan kamera, dan memiliki fungsi dua dimensi yang terbentuk dari penglihatan dalam suatu tempat (*scene)* yang merupakan basis dari pembentukan video. Secara umum video dibagi menjadi dua macam, yaitu :

1. Analog, yaitu video hasil tangkapan lensa kamera terhadap tempat (*scene)* yang di-*scene* secara vertikal dan horizontal oleh *video* kamera.
2. *Digital*, yaitu video yang direpresentasikan sebagai sebuah matriks yang masing-masing elemennya merepresentasikan nilai intensitas

Video *digital* pada dasarnya tersusun atas serangkaian *frame*. Rangkaian *frame* tersebut ditampilkan pada layar dengan kecepatan tertentu , tergantung pada *frame rate* yang diberikan (dalam *frame/second*). Jika *frame rate* cukup tinggi, mata manusia tidak dapat menangkap gambar atau *frame*, melainkan menangkapnya sebagai rangkaian yang kontinu/berlanjut (video).

Masing-masing *frame* merupakan citra *digital*. Suatu citra *digital* direpresentasikan dengan sebuah matriks yang masing-masing elemennya merepresentasikan nilai intensitas. Jika I adalah matriks dua dimensi, I(x,y) adalah nilai intensitas yang sesuai pada posisi baris x dan kolom y pada matriks tersebut. Titik-titik ditempatkan *image* di *sampling* disebut *picture elements*, atau sering dikenal sebagai piksel. *Pixel* atau piksel (*picture element /* unsur gambar*)* adalah tititk–titik kecil. Gambar apapun yang tampak di layar komputer sebenarnya tersusun dari titik-titik kecil.

Jika beberapa piksel diletakkan berderet maka yang tampak adalah sebuah garis. Jadi semua garis, sehalus apapun tampaknya dilayar komputer, sebenarnya adalah deretan piksel. Sebuah piksel memang bisa dianggap sebagai sebuah titik, namun dalam kenyataannya, piksel-piksel lebih mirip dengan persegi panjang kecil yang tingginya tidak sebanding dengan lebarnya.



Gambar 2.5. Piksel A, piksel B dan piksel C (Sumber : Yosi Yanata [11])

2.4.1. Frame Rate

Ketika serangkaian gambar mati yang bersambung dilihat oleh mata manusia, maka suatu keajaiban terjadi. Jika gambar-gambar tersebut dimainkan dengan cepat maka akan terlihat sebuah pergerakan yang halus, inilah prinsip dasar film, video dan animasi. Jumlah gambar yang terlihat setiap detik disebut dengan *frame rate*. Diperlukan frame rate minimal sebesar 10 fps (frame rate per second) untuk menghasilkan gambar pergerakan yang halus. Film-film yang kita lihat di gedung bioskop adalah film yang diproyeksikan dengan frame rate sebesar 24 fps, sedangkan video yang kita lihat di televisi kira-kira memiliki frame rate sebesar 30 fps (tepatnya 29.97 fps) untuk negara yang memakai format standar NTSC (*National Television Standards Comitte*) yaitu Amerika Serikat, Jepang, Kanada, Meksiko dan Korea. Untuk negara Indonesia, Inggris, Australia, Eropa dan China format video standar yang digunakan adalah format PAL (*Phase Alternate Line*) dengan frame rate sebesar 25 fps. Sedangkan negara Perancis, Timur Tengah dan Afrika menggunakan format video standar SECAM (*Sequential Couleur Avec Memoire*) dengan frame rate sebesar 25 fps.

2.4.2. Pixel Ratio

Pixel aspect ratio menjelaskan tentang rasio perbandingan lebar dengan tinggi dari sebuah pixel dalam sebuah gambar. *Frame aspect ratio* menggambarkan perbandingan lebar dengan tinggi pada dimensi frame dari sebuah gambar. Sebagai contoh, D1 NTSC memiliki pixel aspect ratio 0.9 (0.9 lebar dari 1 unit tinggi) dan memiliki pula frame aspect ratio 4:3 (4 unit lebar dari 3 unit tinggi).

Beberapa format video keluaran menggunakan frame aspect ratio yang sama tetapi memakai pixel aspect ratio yang berbeda. Sebagai contoh, beberapa format NTSC digital menghasilkan sebuah 4:3 frame aspect ratio, dengan *square pixel* (1.0 pixel aspect ratio) dan dengan resolusi 640 X 480. Sedangkan D1 NTSC menghasilkan frame aspect ratio yang sama yaitu 4:3 tetapi menggunakan *rectangular pixel* (0.9 pixel aspect ratio) dengan resolusi 720 X 486. Pixel yang dihasilkan oleh format D1 akan selalu bersifat rectangular/bidang persegi, akan berorientasi vertical dalam format NTSC, dan akan berorientasi horizontal dalam format PAL. Jika kita menampilkan rectangular pixel dalam sebuah monitor square pixel tanpa alterasi maka gambar yang bergerak akan berubah bentuk atau mengalami distorsi. Contohnya lingkaran akan berubah menjadi oval. Tetapi bagaimanapun juga apabila ditampilkan pada monitor broadcast gambar gerak akan ditampilkan secara benar.

2.4.3. Resolusi dan Frame Size

Lebar dan tinggi frame video disebut dengan *frame size*, yang menggunakan satuan pixel, misalnya video dengan frame size 640x480 pixel. Dalam dunia digital video, frame size disebut juga dengan resolusi. Semakin tinggi resolusi gambar maka semakin besar pula informasi yang dimuat, berarti akan semakin besar pula kebutuhan *memory* untuk membaca informasi tersebut. Misalnya untuk format PAL D1/DV berukuran 720x576 pixel, format NTSC DV 720x480 pixel dan format PAL VCD/VHS (MPEG-1) berukuran 352x288 pixel sedangkan format NTSC VCD berukuran 320x240 pixel.

2.4.4. Bit Depth

Dalam dunia computer, satuan *bit* merupakan unit terkecil dalam penyimpanan informasi. *Bit depth* menyatakan jumlah/banyaknya bit yang disimpan untuk mendeskripsikan warna suatu pixel. Sebuah gambar yang memiliki 8 bit per pixel dapat menampilkan 256 warna, sedangkan gambar dengan 24 bit dapat menampilkan warna sebanyak 16 juta warna. Komputer (PC) menggunakan 24 bit RGB (Red Green Blue) sedang sinyal video menggunakan standar 16 bit YUV sehingga memiliki jangkauan warna yang terbatas. Untuk itu kita perlu berhati-hati apabila membuat video untuk ditayangkan di TV, karena tampilan warna di layar monitor PC berbeda dengan tampilan di layar TV.

2.4.5. Bit Rate

Bit rate disebut juga dengan nama *data rate*. Bit rate menentukan jumlah data yang ditampilkan saat video dimainkan. Data rate ini dinyatakan dalam satuan bps (bit per second). Data rate berkaitan erat dengan pemakaian dan pemilihan *codec* (metode kompresi video). Beberapa codec menghendaki data rate tertentu, misalnya MPEG-2 yang digunakan dalam format DVD dapat menggunakan bit rate maksimum 9800 kbps atau 9,8 Mbps, sedangkan format VCD hanya mampu menggunakan bit rate 1,15 Mbps.

2.4.6. Redudansi

Di dalam video *digital* mengandung *redundancy* (pengulangan), karena ada data sebenarnya dapat direpresentasikan dengan data lain dalam *frame* yang sama terhadap pergerakan gambar, redudansi dibagi menjadi dua katagori yaitu redundansi *spatial* dan redundansi *temporal.*

1. Redundansi Temporal

   Adalah redundansi yang terjadi pada sebuah *frame* dengan *frame* yang sebelum atau sesudahnya, hal ini dikarenakan *pixel-pixel* tersebut berkorelasi diantara *frame*, redundansi ini dikarenakan banyak bagian *frame* yang tidak berubah dibanding *frame* sebelum dan sesudah nya.

2. Redundansi Spasial

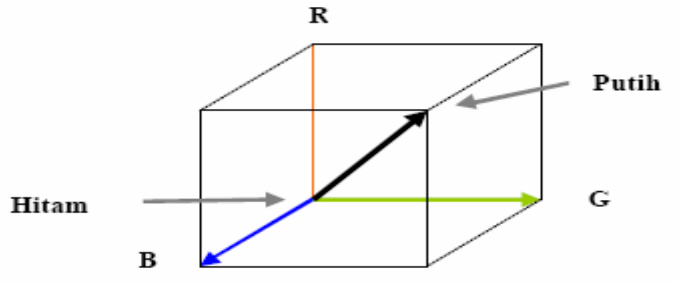
   Merupakan redundansi yang terdapat dalam suatu *frame* yang disebabkan karena adanya korelasi antara sebuah *pixel* dengan *pixel-pixel* disekitarnya

2.4.7. Representasi Warna

Pada video *digital* data dipisahkan menjadi beberapa komponen warna (*crominance*) dan komponen kecerahan, di mana pemisahan tiap komponen menggunakan cara-cara tertentu, beberapa cara pemisahan komponen tersebut adalah RGB, YUV, danYIQ.

1. RGB (*Red, Green, Blue*)

Pada RGB 24 bit, data video dipisahkan ke dalam komponen untuk masing-masing warna merah (*red*), hijau (*green*), biru (*blue*) yang masing-masing memiliki nilai 8 bit atau 256 level.



Gambar 2.6. Koordinat RGB (Sumber : Yonata [11])

Misalnya citra dengan 8 bit per piksel mempunyai 256 warna dan citra dengan 24 bit mempunyai ±16,8 juta warna, jadi tiap piksel dinyatakan dengan:

1. Bit 0 sampai dengan 7 untuk warna merah.
2. Bit 7 sampai dengan 15 untuk warna hijau
3. Bit 16 sampai dengan 24 untuk warna biru

Kemungkinan kombinasi warna yang ada adalah = 2563 + 2562 + 2561 = *16.843.008*, dimana nilai 0 menyatkan warna hitam sedangakan nilai 16 843 008 menyatakan warna putih.

2. YUV (*Yellow Ultra Violet*)

Pemisahan komponen tidak hanya pada warna (*crominance*) tetapi juga terhadap kecerahan (*luminance*) pada format PAL, sinyal kecerahan dinyatakan dalam Y dan masing-masing komponen diperoleh dengan mentransformasikan RGB dengan rumus :

Y = 0,299 R +0,587 G + 0,114 B ……..persamaan (2.1)

U= (B-Y) x 0,493 ……..persamaan (2.2)

V= (R-Y) x 0,877 ……..persamaan (2.3)

3. YIQ (*Yellow Irradiation Quart*)

Pemisahan sinyal video dapat dilakukan juga dalam format NTSC di mana komponen *luminance* dinyatakan dalam Y dan komponen *crominance* dinyatakan dalam I dan Q dengan rumus:

Y = 0,299 R +0,587 G + 0,114 B ……persamaan (2.4)

I= 0,596 R - 0,275 G - 0,321 B ……persamaan (2.5)

Q= 0,212 R – 0,523 G – 0,311 B ……persamaan (2.6)

2.4.8. AVI (*Audio Video Interleave)*

Audio Video Interleave atau dikenal dengan AVI adalah format multimedia yang dikenalkan oleh *Microsoft* pada bulan November 1992, bagian dari teknologi video untuk *windows*, file AVI dapat berisi data video dan audio, sebuah standarisasi yang berisikan sinkronisasi antara audio dan video, seperti DVD file AVI mendukung multi *streaming* audio dan video, disamping digunakan secara

tersendiri, file AVI juga banyak digunakan sebagai format extensi file yang dikembangkan oleh *Matrox Open DML group* pada February 1996.

Pengembangan file AVI yaitu AVI 2.0 digunakan khusus untuk kasus *Resource interchange File Format* (RIFF), sebuah file AVI dapat dikompres kedalam bentuk format kompresi multimedia seperti *Motion JPEG*, *VDOWave*, *ClearVideo/RealVideo*, dan lain-lain serta dapat melakukan *share file* dan melakukan proses *rendering* seperti *capture, edit, playback.* Pada file AVI terdapat beberapa karakteristik dari AVI antara lain :

1. File AVI memiliki masalah dalam ukuran , karena AVI dapat mendukung file yang besar tetapi dalam aplikasinya ada beberapa permasalahan yang terjadi contohnya *player-player* yang mendukung MPEG tidak mendukung file AVI berukuran lebih dari 2 GB.
2. Spesifikasi dari AVI dan kelebihan yang dikembangkan oleh *Matrox Open DML group,* AVI dapat menangani 256 *stream*s baik itu audio ataupun video.
3. AVI dapat menangani subtitle *text*, karena AVI memiliki *AVI-Mux GUI* dan *DVobSub/VSFilter*.
4. *Subtitle* dalam AVI sangat mudah diimplementasikan baik melakukan pemotongan atau penggabungan *subtitle,* blok data dari *subtitle* dapat dilakukan dengan pemrosesan, penampilan, pembuatan, dan perbaikan.
5. AVI tidak mendukung *Bidirectional coding* (*B-frame* pada MPEG), karena AVI tidak dirancang untuk menangani prediksi dari pergerakan *frame* (P-*frame*) dan arah gerak *frame* (*B-frame*) seperti format MPEG.

2.4.9. Steganografi pada Video

Seperti dikatakan sebelumnnya, video merupakan kumpulan dari image yang “bergerak”, jadi sebagian besar metode yang digunakan pada image steganography dapat digunakan pada video steganography. Dapat dikatakan bahwa video steganography merupakan turunan dari image steganography. Pada video steganography ini, menggunakan metode transformasi Fast Fourier Transform. Hal ini dikarenakan modifikasi LSB akan mengasilkan stego yang berukuran sangat besar sedangkan metode masking dan filtering akan mengubah tampilan visual dari video secara langsung.

Untuk menyembunyikan pesan pada cover video, prinsipnya sama seperti pada image steganography. Pertama-tama dilakukan transformasi pada masing-masing frame image cover video untuk memperoleh koefisien-koefisien yang akan dipilih berdasarkan nilai treshold tertentu. Koefisien tersebut akan diganti dengan bit-bit data pesan yang akan disembunyikan. Setelah seluruh pesan di-embed, koefisien tadi ditransformasi balik untuk menghasilkan stego video. Untuk mengekstrak pesan dari stego video, prinsipnya juga sama seperti pada image steganography. Pertama-tama dilakukan transformasi pada masing-masing frame image stego video untuk memperoleh koefisien-koefisien yang akan dipilih berdasarkan nilai treshold tertentu. Koefisien tersebut akan merupakan bit-bit data pesan yang telah disembunyikan dan akan ditulis ke file output yang berisi pesan yang disembunyikan tersebut.

Keuntungan dari video steganography adalah banyaknya data yang dapat disembunyikan di dalamnya, serta fakta bahwa video merupakan “streams” dari image-image menyebabkan adanya distorsi pada salah satu frame image tidak akan dilihat dengan mudah dengan mata manusia. Akan tetapi, semakin banyak data pesan yang disembunyikan, bukan hal yang mustahil jika perubahan pada video menjadi semakin mudah terlihat.

2.5. Teknik Transformasi

Pengkodean transform merupakan sebuah komponen *integral* dari aplikasi pemerosesan *image*/video. Pengkodean transform mempercayakan *pixel* dalam tampilan *image* pada tingkatan *pixel* yang berkorelasi dengan *pixel* tetangganya, umumnya sebuah sistem transmisi video, *pixel* terdekat dalam *frame* secara berurutan menampilkam korelasi yang tinggi, akibatnya korelasi dapat ditampilkan pada prediksi nilai sebuah *pixel* dari masing-masing *pixel* tetangga.

2.5.1. FFT (*Fast Fourier Transform*)

Citra pada umumnya digambarkan dalam dimensi spasial antara amplitude dengan posisi spasial dua dimensi. Dalam operasi analisis citra, seringkali dilakukan operasi konvolusi pada matriks (citra) yang berdimensi besar. Hal ini menjadi salah satu penghambat karena ketika dilakukan operasi konvolusi, maka dibutuhkan resource computer yang sangat besar

Agar operasi analisis citra dapat dilakukan dengan efisien, maka sebaiknya citra dirubah dahulu kedalam dimensi frekuensi. Hal ini sangat penting dilakukan

terlebih dalam operasi yang mengunakan konvolusi karena konvolusi dalam dimensi spasial berarti perkalian produk (dot product) dalam dimensi frekuensi. mempercepat waktu proses. Untuk melakukan transformasi dari dimensi spasial menjadi dimensi frekuensi, salah satunya adalah dengan menggunaka transformasi Fourier. Inti dari transformasi Fouries adalah memecah signal (citra) menjadi gelombang-gelombang sinusoidal dimana jumlahnya sama dengan signal asalnya.

Dalam fungsi yang kontinyu, transformasi Fourier dinyatakan dalam bentuk sebagai berikut:

$$\Im\{f(x)\} = F(u) = \int_{-\infty}^{\infty} f(x) e^{1i2\pi ux} du$$

Dan fungsi balikannya adalah sebagai berikut

$$\Im\{F(u)\} = f(x) = \int_{-\infty}^{\infty} F(u) e^{1i2\pi ux} du$$

Dalam konteks dua dimensi, fungsi Fourier menjadi seperti berikut:

$$\Im\{f(x,y)\} = F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) e^{1i2\pi ux} du$$

Dan fungsi balikannya adalah sebagai berikut:

$$\Im\{F(u,v)\} = f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v) e^{1i2\pi ux} du$$

Untuk F(u,v) bilangan kompleks, F(u,v) dapat dinyatakan sebagai berikut:

$$F(u,v) = R(u,v) + iI(u,v) = |F(u,v)| e^{i\phi(u)}$$

Sehingga spektrumnya dapat dinyatakan sebagai:

$$|F(u,v)| = \sqrt{R^2(u,v) + i\,I^2(u,v)}$$

Karena citra bukan merupakan signal yang kontinyu, maka fungsi transformasi Fourier diatas harus dinyatakan dalam bentuk diskrit juga.

Fungsi fourier dalam bentuk diskrit disebut juga Discrete Fourier Transform (DFT).

Jika f(x) yang kontinyu dibuat menjadi diskrit dengan mengambil N buah sampel berjarak Δx, yaitu himpunan nilai

$$\{f(x_0), f(x_0 + \Delta x), f(x_0 + 2\Delta x), ..., f(x_0 + (N-1)\Delta x)\}$$

Jadi

$$f(x_0 + x\Delta x), x = 0,1,2,3,..., N-1$$

Sehingga dalam bentuk diskrit, fungsi Fourier bervariabel dua menjadi sebagai berikut:

$$F_{x,y} = \frac{1}{NM} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} F_{u,v} e^{-i2\pi(ux/N + vy/M)}$$

Sedangkan fungsi balikannya adalah sebagai berikut:

$$f_{x,y} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} F_{u,v} e^{i2\pi(ux/N + vy/M)}$$

Algoritma DFT standar tidak bagus jika digukanan pada citra berukuran besar, karena proses komputasinya memakan waktu yang lama. Untuk mempercepat proses transformasi, digunakan algoritma FFT (Fast Fourier Transform). Algoritma ini sangat cepat mengerjakan matriks yang berukuran besar. Perbedaan kompleksitas waktu DFT dengan FFT adalah

$$0(N^2) : 0(N^2 \log N)$$

Dengan FFT proses komputasi biasa direduksi dari N2 menjadi N log2N. Misalnya dengan menggunakan DFT kita akan melakukan transformasi sebanyak N=1024 titik, maka kita memerlukan perkalian sebanyak N2 =1.048.567. Sedangkan dengan menggunakan FFT perkalian yang diperlukan sebanyak N log2N = 5120 perkalian.

2.6. LSB *(Least Significant Bit)*

LSB (Least Significant Bit) Coding. Metoda ini merupakan metoda yang sederhana. Metoda ini akan mengubah nilai LSB (*Least Significant Bit*) komponen luminansi atau warna menjadi *bit* yang bersesuai dengan *bit* label yang akan disembunyikan. Memang metoda ini akan menghasilkan video rekontruksi yang sangat mirip dengan aslinya, karena hanya mengubah nilai *bit* terakhir dari data. Metoda ini paling mudah diserang, karena bila orang lain tahu maka tinggal membalikkan nilai dari LSB-nya maka data label akan hilang seluruhnya.

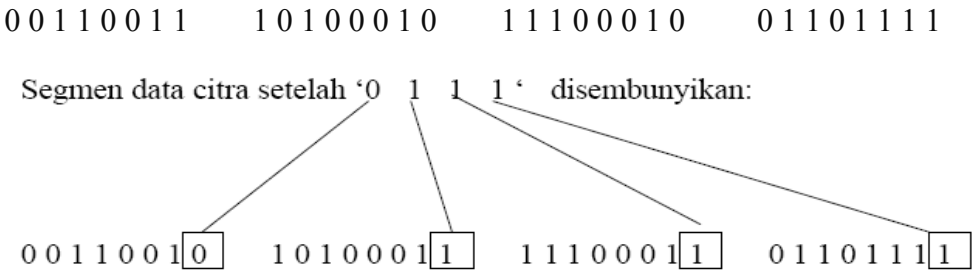
1. Penyembunyian data dilakukan dengan mengganti *bit-bit* data di dalam segmen citra dengan *bit-bit* data rahasia. Metode yang paling sederhana adalah metode **LSB** (*Least Significant Bit Modification*).
2. Pada susunan *bit* di dalam sebuah *byte* (1 *byte* = 8 *bit*), ada *bit* yang paling berarti (*most significant bit* atau *MSB*) dan *bit* yang paling kurang berarti (*least significant bit* atau *LSB*).
3. Perhatikan contoh sebuah susunan *bit* pada sebuah *byte*:

   MSB → 1101001**0** ← *LSB* *LSB = Least Significant Bit*

   MSB = Most Significant Bit

Bit yang cocok untuk diganti adalah *bit LSB*, sebab perubahan tersebut hanya mengubah nilai *byte* satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya. Misalkan 6 *byte* tersebut menyatakan warna merah, maka perubahan satu *bit* LSB tidak mengubah warna merah tersebut secara berarti. Lagi pula, mata manusia tidak dapat membedakan perubahan yang kecil.

4. Misalkan segmen data citra sebelum perubahan:



5. Untuk memperkuat teknik penyembunyian data, *bit-bit* data rahasia tidak digunakan mengganti *byte-byte* yang berurutan, namun dipilih susunan *byte* secara acak. Misalnya jika terdapat 50 *byte* dan 6 *bit* data yang akan disembunyikan, maka maka *byte* yang diganti *bit LSB*-nya dipilih secara acak, misalkan *byte* nomor 36, 5, 21, 10, 18, 49.
6. Bilangan acak dapat dibangkitkan dengan program *pseudorandom- number-generator* (*PRNG*). *PRNG* menggunakan kunci rahasia untuk membangkitkan posisi *pixel* yang akan digunakan untuk menyembunyikan *bit-bit*.
7. *PRNG* dibangun dalam sejumlah cara, salah satunya dengan menggunakan algoritma kriptografi berbasis blok (*block cipher*). Tujuan dari enkripsi adalah menghasilkan sekumpulan bilangan acak yang sama untuk setiap kunci enkripsi yang sama. Bilangan acak dihasilkan dengan cara memilih *bit-bit* dari sebuah blok data hasil enkripsi.