

BAB II

LANDASAN TEORI

2.1 Area parkir

Meningkatnya jumlah kendaraan pribadi saat ini khususnya mobil membuat kebutuhan lahan parkir yang terbuka yang luas. Tidak mudah untuk mencari ruang parkir yang kosong di tempat parkir yang terbuka, hal ini disebabkan karena ruang parkir yang tersedia sudah oleh pengendara mobil lainya, sehingga pengemudi yang ingin menggunakan lahan parkir harus mengitari area parkir tersebut sampai mendapatkan ruang yang kosong, dan banyak juga pengemudi yang memarkirkanya secara paralel sehingga tidak pada ruang yang tersedia. (Muhadi Ardhie Tyasa, 2013).

Kurangnya informasi di area parkir mobil membuat para pengendara yang ingin memarkirkan kendaraanya mencaji sangat kerepotan. Sebenarnya masalah tersebut dapat diatasi dengan memanfaatkan teknologi komputer. Yang saat ini telah dikembangkan beberapa teknologi tempat parkir hanya dengan berupa petunjuk apakah tempat parkir tersebut penuh atau kosong pada papan yang terletak pada pintu masuk parkir. (Muhadi Ardhie Tyasa, 2013)

2.2 VCR (Video Capture Recorder)

(Kurniawan, 2009) Video capture adalah sebuah proses digitalisasi dari video analog ke dalam komputer dari perangkat di luar komputer seperti siaran TV dari TV tuner, VCR, kamera dan lain sebagainya, proses ini mendefinisikan

gambar analog menjadi elemen – elemen dasar gambar digital, elemen gambar digital adalah :

1. Kecerahan (*brightness*)
2. Kontras (*contrast*)
3. Kontur (*contour*)
4. Warna (*color*)
5. Bentuk (*shape*)

2.3 Pengolahan Citra

Pengolahan citra adalah memanipulasi dan analisis suatu informasi gambar yang dilakukan oleh komputer, informasi gambar di sini adalah gambar visual yang dibentuk dalam dua dimensi. Tujuan dari pengolahan citra sendiri adalah untuk memperbaiki informasi yang terdapat pada gambar agar dapat diterima secara maksimal dan mudah terbaca, dan untuk memperbaiki kualitas dari gambar yang diterima.

Pengolahan citra memiliki beberapa jenis pengoprasian, yaitu sebagai berikut.

1. Perbaiki kualitas citra (*image enhacement*).

Jenis operasi ini bertujuan untuk memperbaiki kualitas citra dengan cara memanipulasi parameter-parameter citra.

Dengan operasi ini, ciri-ciri khusus yang terdapat di dalam citra lebih ditonjolkan. Contoh-contoh operasi perbaikan citra:

- a. perbaikan kontras gelap/terang
- b. perbaikan tepian objek (*edge enhancement*)
- c. penajaman (*sharpening*)

d. *grayscale*

e. *noise filtering*

2. Pemugaran citra (*image restoration*)

Operasi ini bertujuan menghilangkan atau meminimumkan cacat pada citra.

Tujuan pemugaran citra hampir sama dengan operasi perbaikan citra.

Bedanya, pada pemugaran citra penyebab degradasi gambar diketahui.

Contoh-contoh operasi pemugaran citra:

a. penghilangan kesamaran (*deblurring*)

b. penghilangan *noise*

3. Penempatan citra (*image compression*)

Jenis operasi ini dilakukan agar citra dapat direpresentasikan dalam bentuk yang lebih kompak sehingga memerlukan memori yang lebih sedikit. Hal penting yang harus diperhatikan dalam pemampatan adalah citra yang telah dimampatkan harus tetap mempunyai kualitas gambar yang bagus. Contoh metode pemampatan citra adalah metode JPEG.

4. Segmentasi citra (*image segmentation*)

Jenis operasi ini bertujuan untuk memecah suatu citra ke dalam beberapa segmen dengan suatu kriteria tertentu. Jenis operasi ini berkaitan erat dengan pengenalan pola.

5. Pengorakan citra (*image analysis*)

Jenis operasi ini bertujuan menghitung besaran kuantitatif dari citra untuk menghasilkan deskripsinya. Teknik pengorakan citra mengekstraksi ciri-ciri tertentu yang membantu dalam identifikasi objek. Proses segmentasi

kadangkala diperlukan untuk melokalisasi objek yang diinginkan dari sekelilingnya. Contoh-contoh operasi pengolahan citra:

- a. Pendeteksian tepi objek (*edge detection*)
 - b. Ekstraksi batas (*boundary*)
 - c. Representasi daerah (*region*)
6. Rekanstruksi citra (*image reconstruction*)

Jenis operasi ini bertujuan untuk membentuk ulang objek dari beberapa citra hasil proyeksi. Operasi rekonstruksi citra banyak digunakan dalam bidang medis. Misalnya beberapa foto *rontgen* dengan sinar X digunakan untuk membentuk ulang gambar organ tubuh. (Nixon dan Aguado, 2002)

Citra digital (diskrit) dihasilkan dari citra analog (*kontinu*) melalui digitalisasi. Digitalisasi citra analog terdiri *sampling* dan *quantization*. *Sampling* adalah pembagian citra ke dalam elemen-elemen diskrit (*pixel*), sedangkan *quantization* adalah pemberian nilai intensitas warna pada setiap *pixel* dengan nilai yang berupa bilangan bulat (Awcock, 1996). Beberapa hal yang perlu diperhatikan dalam pengolahan citra digital adalah kuantitas data, pendekatan statistik, kecepatan, keefisienan, sistematika, pencarian informasi kembali (*retrieval*), dan modifikasi, termasuk resolusi citranya. (Purwadhi, 2001)

2.4. Brightness

Brightness adalah kata lain untuk intensitas cahaya. Kecerahan pada sebuah titik (*pixel*) di dalam citra bukanlah intensitas yang *real*, tetapi sebenarnya adalah intensitas rata – rata dari suatu area yang mencakupnya. Sistem visual manusia mampu menyesuaikan dirinya dengan tingkat kecerahan (*brightness level*) mulai

dari yang paling rendah sampai yang paling tinggi dengan jangkauan sebesar 10^{10} . (Fatmawati, 2011)

Untuk membuat citra lebih terang atau lebih gelap, kita dapat melakukan pengubahan kecerahan citra. Kecerahan atau kecemerlangan citra dapat diperbaiki dengan menambahkan atau mengurangi sebuah konstanta kepada dari setiap *pixel* didalam citra. Akibat operasi ini, histogram mengalami pergeseran. (Munir, 2004)

Nilai *pixel* hasil pengubahan mungkin lebih kecil atau sama dengan derajat keabuan minimum (0) atau lebih besar sama dengan derajat keabuan maksimum (255). Karena itu, *pixel* tersebut perlu dilakukan *clipping* ke nilai keabuan minimum atau kenilai keabuan maksimum. (Munir, 2004)

2.5. Contrast Streching

Contrast streching biasa disebut dengan normalisasi mencoba untuk meningkatkan kualitas gambar dengan peregangan dengan intensitas nilai dalamnya agar dapat memanfaatkan sepenuhnya kemungkinan nilai yang didapatkan oleh gambar. *Contrast streching* biasa digunakan untuk mendapatkan citra baru dengan kontras yang lebih baik daripada kontras asalnya. (Dharmawan, 2013)

Contrast streching digunakan untuk meningkatkan jangkauan dinamis dari tingkat abu-abu dalam gambar. Sebagai contoh, dalam sistem 8-bit tampilan gambar dapat menunjukkan maksimum 256 tingkat abu-abu. (Dharmawan, 2013)

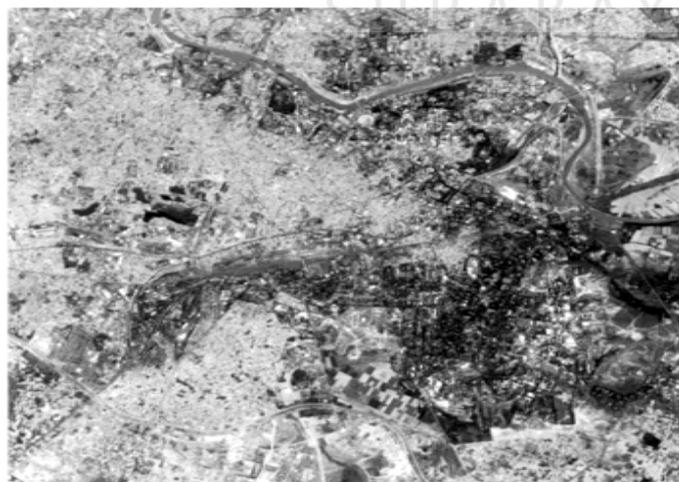
Citra yang memiliki kontras rendah dapat terjadi karena kurangnya pencahayaan dalam gambar, kurangnya bidang dinamika dari sensor citra, atau kesalahan pdalam menggada pengaturan saat pembukaan lensa dalam mengambil

sebuah citra. Ide dari proses *contras stretching* adalah untuk meningkatkan bidang dinamika dari *gray level* di dalam citra yang akan diproses. Citra dengan kontras rendah mempunyai kualitas yang relatif kurang baik, sehingga informasi yang diterima sulit untuk diinterpretasikan oleh mata manusia. (Dharmawan, 2013)

Gambar pengindraan jauh diperoleh dari *multi-spectral*. FCC (*False Color Composite*) adalah kombinasi dari ketiga *band*, seperti di mana masing – masing band diberi tugas untuk pewarnaan primer yang berbeda, yaitu (RGB) *Red, green, dan blue*. (Dharmawan, 2013)



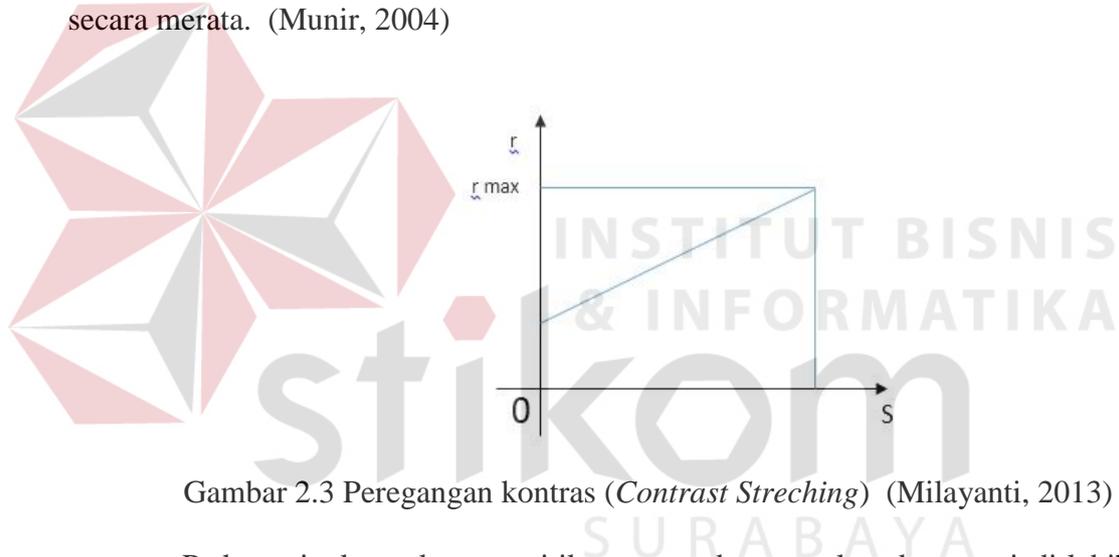
Gambar 2.1 Sebelum proses kontras stretching (Kumar, 2010)



Gambar 2.2 Setelah pemrosesan pada kontras stretching (Kumar, 2010)

Sering kali gambar dipindai sedemikian hingga, untuk mengubah kecerahan yang dihasilkan tidak menggunakan rentang dinamis yang sedemikian hingga. Penyempurnaan kontras ditingkatkan agar daya nalar benda pada gambar dapat ditingkatkan kecerahannya antara obyek dan *background* nya. (Eschbach, 2009).

Citra kontras rendah dapat diperbaiki kualitasnya dengan operasi peregangan kontras. Melalui operasi ini, nilai warna keabuan akan merentang dari 0 sampai 225 (pada citra 8-bit), dengan kata lain nilai warna keabuan terpakai secara merata. (Munir, 2004)



Gambar 2.3 Peregangan kontras (*Contrast Stretching*) (Milayanti, 2013)

Pada peningkatan kontras, titik yang cenderung gelap akan menjadi lebih gelap, dan yang cenderung terang akan menjadi lebih cemerlang. Peningkatan kontras juga bisa dilakukan dengan berbagai macam rumus, salah satunya : (Milayanti, 2013)

$$S = hl - Q \dots\dots\dots(1)$$

Kontras menyatakan sebaran terang (*lightness*) dan gelap (*darkness*) di dalam sebuah gambar. Citra dengan kontras rendah di cirikan oleh sebagian besar komposisi citranya adalah terang atau sebagian besar citranya gelap. Pada citra

dengan kontras yang baik, komposisi citra gelap dan citra terang tersebar secara merata. (Fatmawati, 2011)

Metode pada *contrast stretching* sangat tergantung dari kebutuhan, apakah kontras citra mau dibuat kearah gelap atau yang terang. *Contrast* yang baik akan terlihat pada nilai – nilai grafik histogram yang menyebar secara merata dari warna gelap hingga warna terang. (Choerudin, 2013)

Cara kerja dari proses *contrast stretching* (peregangan kontras) pada citra 8-bit adalah sebagai berikut.

1. Mengelompokan dan menghitung *pixel* berdasarkan nilai keabuan (membuat histogram). (Choerudin, 2013)
2. Memindai (*Scan*) histogram dari nilai keabuan terkecil ke nilai keabuan terbesar (0 hingga 255) untuk menemukan batas nilai keabuan yang terendah dan batas nilai keabuan tertinggi dari kelompok *pixel* (citra). (Choerudin, 2013)
3. Memetakan (menskalakan) berdasarkan *pixel pixel* yang berada di antara batas terendah hingga batas tertinggi untuk memenuhi rentang nilai dari warna keabuan yang lengkap (0 hingga 255) dengan persamaan: (Choerudin, 2013)

$$Y = \frac{X - x_1}{X_2 - x_1} * 255 \dots\dots\dots(2)$$

Dalam hal ini Y adalah nilai keabuan yang baru atau diperoleh, X adalah nilai keabuan yang lama, X_1 adalah nilai keabuan tertinggi dari kelompok *pixel* dan X_2 adalah nilai keabuan terendah dari kelompok *pixel*. Dengan menggunakan persamaan (2) maka akan didapatkan nilai keabuan 0-255. (Choerudin, 2013)

Metode yang digunakan dalam *contrast stretching* adalah dengan menghitung *pixel* dari *image* dan kemudian melakukan proses untuk melebarkan *pixel* levelnya : (sembiring, 2012)

$$\frac{\text{Pixel lama} - \text{pixel level terendah}}{\text{Pixel level tertinggi} - \text{pixel level terendah}} \times 255 \dots\dots\dots(3)$$

2.6. Grayscale

Proses awal yang sering dilakukan dalam melakukan *image processing* adalah mengubah citra berwarna menjadi citra *grayscale*. Hal ini digunakan untuk menyederhanakan model citra. (Dharmawan, 2013). Citra *grayscale* hanya menggunakan warna satu tingkatan warna abu – abu. . Warna abu – abu adalah satu – satunya warna pada ruang RGB (Red, Green, Blue) yang mempunyai intensitas warna yang sama.

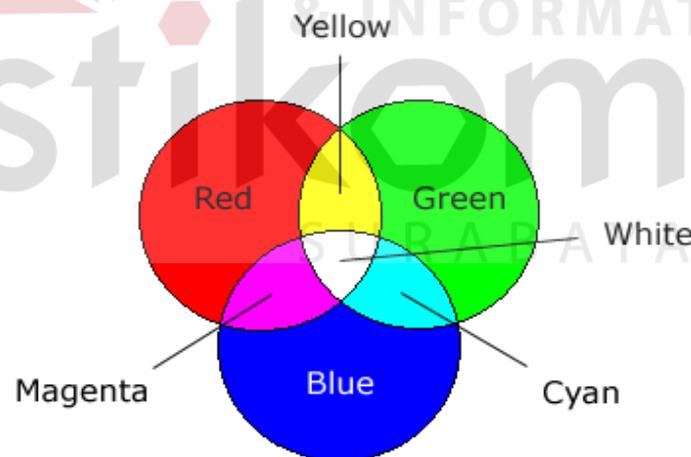
Pada citra berbasis keabu–abuan hanya memerlukan nilai intensitas untuk tiap piksel sebagai nilai tunggal, sedangkan pada citra yang berwarna memerlukan sebanyak tiga nilai intensitas untuk setiap pikselnya. Intensitas dari citra berwarna grayscale akan disimpan ke dalam 8 bit integer yang memberikan 256 kemungkinan yang mana akan dimulai dari level 0 hingga 255 (0 untuk warna hitam dan 255 untuk warna putih dan di antara 0 – 255 adalah derajat warna keabuan). (sunu jatmika, 2014)



Gambar 2.4 Grayscale level (sunu jatmika, 2014)

2.7. RGB (Red, Green, Blue)

RGB mengacu pada sistem untuk mewakili warna yang akan digunakan pada layar komputer. Warna merah, hijau, dan biru dapat dikombinasikan dalam berbagai proporsi untuk mendapatkan warna spektrum. Tingkat R, G, dan B mendapat nilai rentang dari 0 sampai 100 persen dari intensitas penuh. (Rouse, 2005).



Gambar 2.5 Contoh representasi RGB (Pradhitya, 2015)

Pengolahan warna dengan menggunakan model RGB sangat mudah dan sederhana, karena informasi warna dalam komputer sudah dikemas dalam model yang sama. Hal yang perlu dilakukan adalah melakukan pembacaan nilai – nilai R, G, dan B pada satu *pixel*. (Pradhitya, 2015)

Salah satu cara untuk menghitung warna yaitu dengan cara menormalisasi ketiga komponen warna tersebut. Normalisasi penting dilakukan terutama bila sejumlah citra ditangkap dengan penerangan yang berbeda – beda. Hasil perhitungan tiap komponen warna pokok yang telah dinormalisasi akan menghilangkan pengaruh penerangan sehingga setiap komponen warna dapat dibandingkan satu sama lainnya walaupun berasal dari citra dengan kondisi penerangan yang tidak sama. (Pradhitya, 2015)

Cara menormalisasi adalah sebagai berikut.

$$r = \frac{R}{R + G + B} \dots\dots\dots(4)$$

$$g = \frac{G}{R + G + B} \dots\dots\dots(5)$$

$$b = \frac{B}{R + G + B} \dots\dots\dots(6)$$

Bila ketiga warna komponen warna yang telah dinormalisasi dengan masing – masing indeks warna merah (r), indeks warna hijau (g), dan indeks berwarna biru (b), mempunyai nilai yang sama (1/3), maka objek tidak berwarna. Bila (r) lebih besar dari pada g dan b, maka objek berwarna merah, dan seterusnya. (Pradhitya, 2015)

2.8. Histogram

Histogram adalah grafik yang menggambarkan penyebaran dari nilai – nilai *pixel* yang terdapat pada ssuatu citra atau bagian citra tertentu didalamnya. Intensitas pada citra dapat diketahui dari sebuah histogram. Histogram juga dapat mengetahui tentang banyak hal tentang kecerahan (*Brigrthnets*) dan kontras

(Contrast) dari sebuah gambar. Karena itu histogram adalah alat bantu yang sangat berharga dalam sebuah pengerjaan pengolahan citra, baik secara kuantitatif ataupun kualitatif. (Fatmawati, 2011)

Histogram merupakan dasar untuk berbagai teknik pengolahan domain spesial. Manipulasi histogram dapat digunakan secara efektif untuk peningkatan citra, seperti yang ditampilkan di bagian ini. Histogram sederhana untuk menghitung dalam perangkat lunak dan juga untuk implementasi hardware, sehingga membuat mereka populer dalam pengolahan citra yang *real-time*. (Rafael C. Gonzalez, 2004)

Penyebaran nilai intensitas harus diubah supaya bisa menghasilkan hasil citra yang baik, dan teknik yang biasa dipakai adalah pemerataan histogram (*Histogram equalization*). Tujuan dari pemerataan histogram sendiri adalah untuk memperoleh penyebaran yang merata, sehingga setiap derajat keabuan akan memiliki jumlah pixel yang relatif sama. (Fatmawati, 2011)

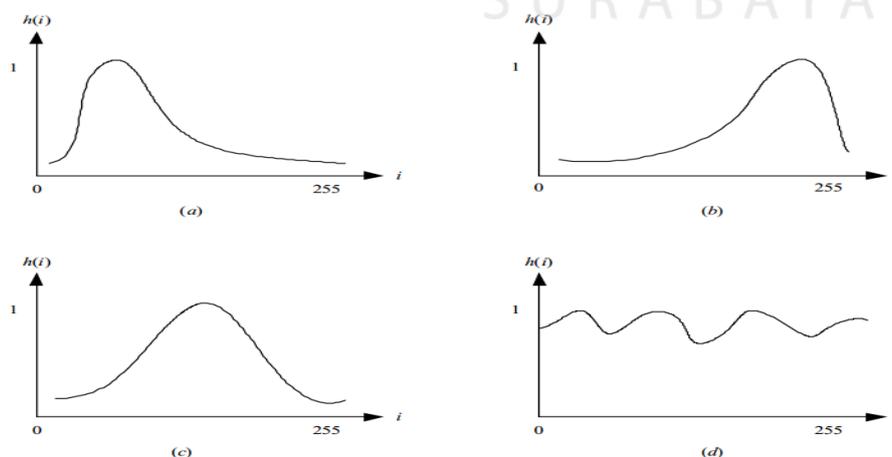
Membandingkan histogram dari citra mentah dan citra yang ditingkatkan menggunakan linear *stretching* dan histogram yang disamakan. Apa saja bagian yang terdapat pada histogram? (Harrison, 2005)

1. Judul, judul singkat yang menjelaskan tentang isi yang terkandung dalam histogram. (Harrison, 2005)
2. Horizontal atau X-Axis: horizontal atau sumbu X berisi skala yang memiliki nilai – nilai yang pengukurannya cocok. Pada umumnya pengukuran ini dikelompokkan kedalam interval untuk membantu meringkas set data yang besar, sedangkan data individu tidak ditampilkan. (Harrison, 2005)

3. Bar: Bar memiliki dua karakteristik yang penting, yaitu tinggi dan lebar. Tinggi untuk menunjukkan jumlah nilai dalam selang waktu yang terjadi, sedangkan *width* merupakan panjang pada interval yang tertutup oleh bar. (Harrison, 2005)
4. Vertical or Y-Axis: vertikal atau Sumbu Y adalah skala yang berisi beberapa waktu nilai yang terdapat pada selang waktu yang terjadi. Jumlah waktu juga biasa disebut sebagai frekuensi. (Harrison, 2005)
5. Legend : *legend* memberikan informasi tambahan bahwa dokumen mana data itu berasal dari dan bagaimana pengukuran dikumpulkan. (Harrison, 2005)

2.8.1. Pembuatan Histogram

1. Ciri – ciri citranya
 - a. Gelap
 - b. Terang
 - c. Normal
 - d. Normal *Brightness* dan *Contrast*



Gambar 2.6 (a) Citra Gelap, (b) Citra terang, (c) Citra normal (d) normal

brightness dan *high contrast* (Prof. Dr.rer.nat. Achmad Benny Mutiara, 2005)

Gambar 7. Memperlihatkan tiga buah citra lena. Citra lena yang pertama terlalu gelap. Histogramnya banyak menumpuk pada bagian kiri, karena citra tersebut mengandung banyak nilai intensitas yang dekat dengan 0 (hitam). Citra lena yang kedua terlalu terang. Histogramnya banyak menumpuk pada bagian kanan, karena citra tersebut mengandung banyak nilai intensitas yang dekat dengan 255 (putih). Citra lena yang ketiga adalah citra yang normal (bagus). Histogram tersebar merata di seluruh daerah derajat keabuan. (Prof. Dr.rer.nat. Achmad Benny Mutiara, 2005)

2.8.2. Membuat Histogram

Misalkan citra digital memiliki L derajat keabuan, yaitu dari nilai 0 sampai $L-1$ (misalnya pada citra dengan kuantitas derajat keabuan 8-bit, nilai derajat keabuan dari 0 sampai 255). Secara sistematis histogram citra dihitung dengan rumus

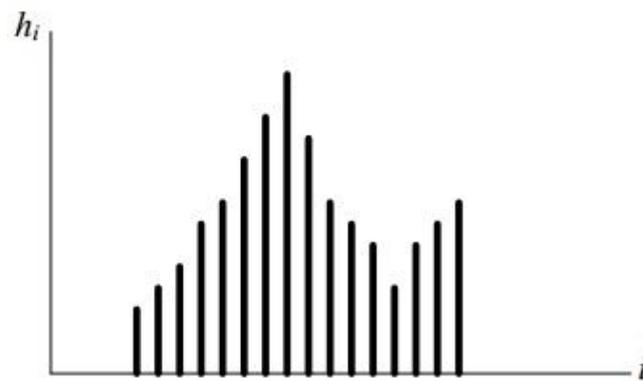
$$h_i = \frac{n_i}{n} \quad i = 0, 1, \dots, L-1 \quad \dots \dots \dots (7)$$

Yang dalam hal ini,

n_i = jumlah *pixel* yang memiliki derajat keabuan i

n = jumlah seluruh *pixel* di dalam citra

Plot h_i versus f_i dinamakan histogram. Gambar 8. Adalah contoh sebuah histogram citra. Secara grafis histogram ditampilkan dengan diagram batang. Dapat diperhatikan bahwa persamaan (7), nilai n_i telah dinormalkan dengan membaginya dengan n , nilai h_i berada di dalam selang 0 sampai 1. (Prof. Dr.rer.nat. Achmad Benny Mutiara, 2005)



Gambar 2.7 Histogram citra (Prof. Dr.rer.nat. Achmad Benny Mutiara, 2005)

Fitur yang dimiliki OpenCV antara lain :

1. Manipulasi data citra (*allocation, copying, setting, convert*).
2. Citra dan video I/O (*file dan kamera based input, image/video file output*).
3. Manipulasi Matriks dan Vektor beserta rutin-rutin aljabar linear (*products, solvers, eigenvalues, SVD*).
4. Data struktur dinamis (*lists, queues, sets, trees, graphs*).
5. Pemroses citra fundamental (*filtering, edge detection, corner detection, sampling and interpolation, color conversion, morphological operations, histograms, image pyramids*).
6. Analisis struktur (*connected components, contour processing, distance Transform, various moments, template matching, Hough Transform, polygonal approximation, line fitting, ellipse fitting, Delaunay triangulation*).
7. Kalibrasi kamera (*calibration patterns, estimasi fundamental matrix, estimasi homography, stereo correspondence*).
8. Analisis gerakan (*optical flow, segmentation, tracking*).
9. Pengenalan obyek (*eigen-methods, HMM*).

10. Graphical *User Interface* (*display image/video*, penanganan *keyboard* dan *mouse handling*, *scroll-bars*).

OpenCV terdiri dari 3 *library*, yaitu:

1. CV : Untuk algoritma *Image Processing* dan *Vision*
2. Highgui : Untuk GUI, *Image* dan *Video I/O*
3. CXCORE : Untuk struktur data, *support XML* dan fungsi-fungsi grafis.

