



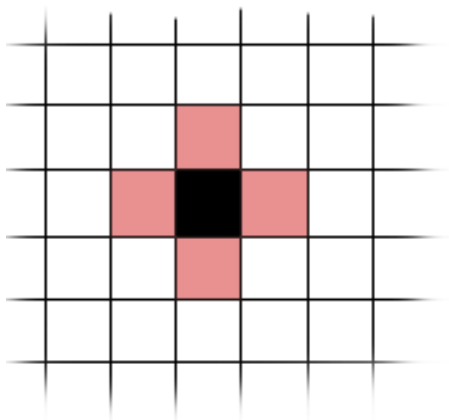
Peningkatan Kualitas Citra (Image Enhancement)

Hubungan antar pixel (1/4)

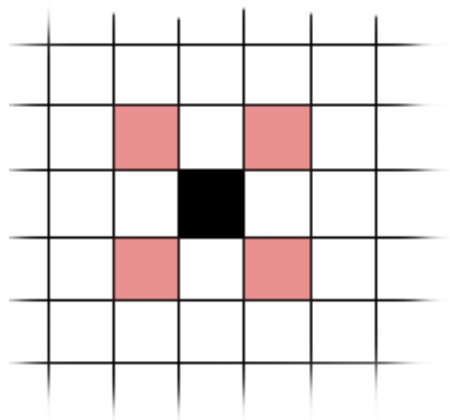
- **Neighbourhood**
- **Connectivity**
- **Distance**

Hubungan antar pixel (2/4)

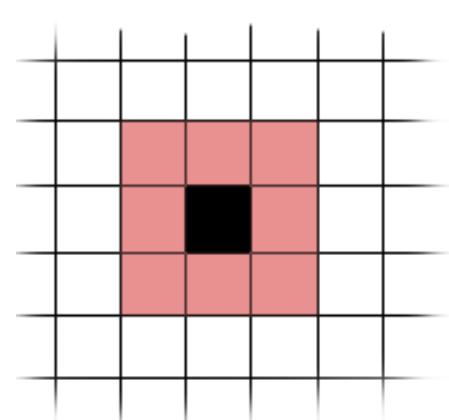
- **Neighbourhood (tetangga pixel)(1)**
 - Tetangga horisontal dan vertikal → $N_4(p)$
 - Tetangga diagonal → $N_D(p)$
 - 8-tetangga → $N_8(p)$



$N_4(p)$
)



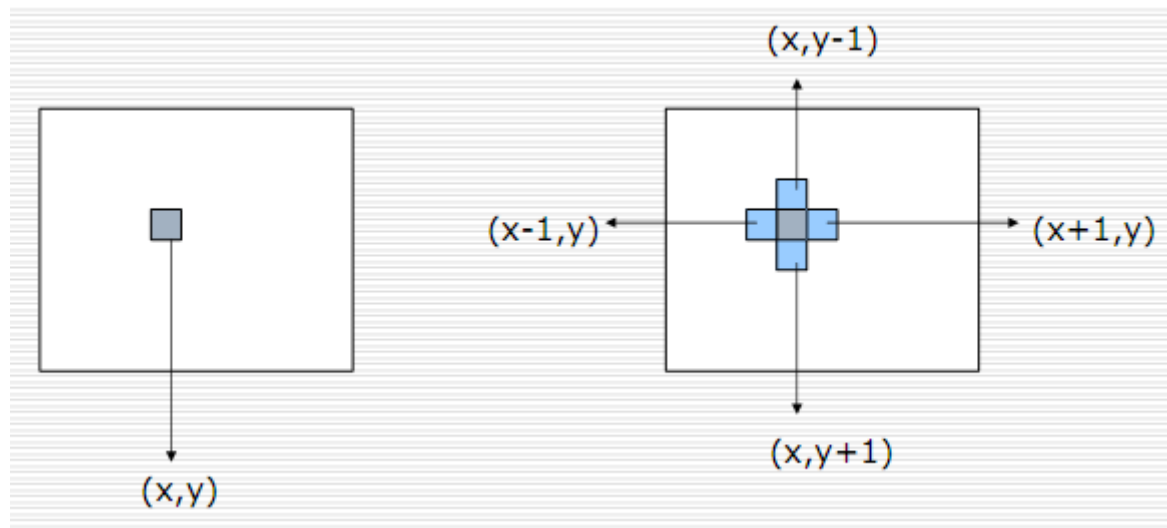
$N_D(p)$
)



$N_8(p)$
)

Hubungan antar pixel (2/4)

- **Neighbourhood (tetangga pixel)(2)**
- **Tetangga horisontal dan vertikal $\rightarrow N_4(p)$**
 - 4 titik tetangga (x,y) adalah titik-titik: $(x-1,y)$, $(x+1,y)$, $(x,y-1)$ dan $(x,y+1)$ sebagai tetangga kiri, kanan, atas dan bawah

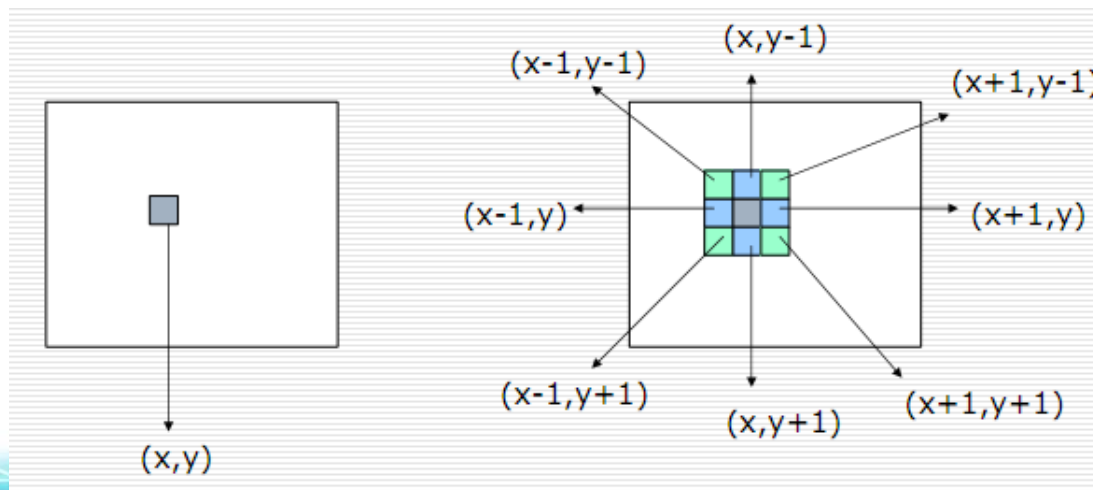


Hubungan antar pixel (2/4)

□ Neighbourhood (tetangga pixel)(3)

□ 8-tetangga $\rightarrow N_8(p)$

- Titik (x,y) dan 8 titik tetangganya merupakan suatu matrik ukuran 3×3 yang merupakan dasar dari pengolahan citra lebih lanjut.
- 8 titik tetangga (x,y) adalah titik-titik: $(x-1,y-1)$, $(x-1,y)$, $(x-1,y+1)$, $(x,y-1)$, $(x,y+1)$, $(x+1,y-1)$, $(x+1,y)$ dan $(x+1,y+1)$



Hubungan antar pixel (3/4)

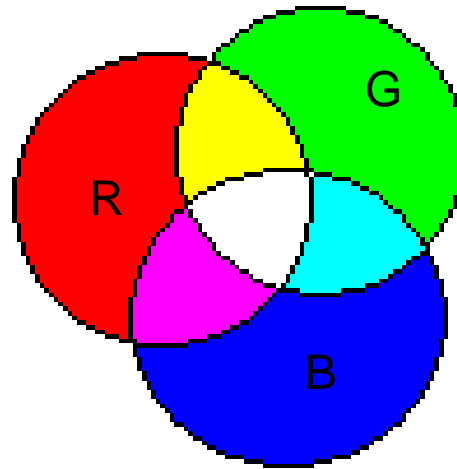
- **Connectivity (1)**
 - Menentukan apakah 2 pixel saling berhubungan berdasarkan kriteria tertentu
 - Merupakan konsep penting untuk menentukan batas objek
 - **Syarat konektivitas adalah :**
 - 2 pixel memiliki gray level yang hampir sama
 - 2 pixel tersebut bertetangga
 - **Contoh kriteria gray level :**
 - Misalkan pada suatu image **8 bit (warna =256)**, konektivitas terjadi bila kedua pixel terletak pada himpunan $V=(32,33,34,...,62,63)$

Hubungan antar pixel (4/4)

- **Connectivity (2)**
 - **3 macam konektivitas :**
 - **4-konektivitas**
 - Dua pixel p dan q dengan gray level termasuk dalam V bila q adalah anggota himpunan **$N_4(p)$**
 - **8-konektivitas**
 - Dua pixel p dan q dengan gray level termasuk dalam V bila q adalah anggota himpunan **$N_8(p)$**
 - **M-konektivitas (konektivitas campur)**
 - Dua pixel p dan q dengan gray level termasuk dalam V bila
 - i. q adalah anggota himpunan $N_4(p)$ atau
 - ii. q adalah anggota himpunan $N_D(p)$ dan $N_4(p) \cap N_4(q) = \emptyset$










□ (pengecekan i dilakukan lebih dulu)

Format Warna

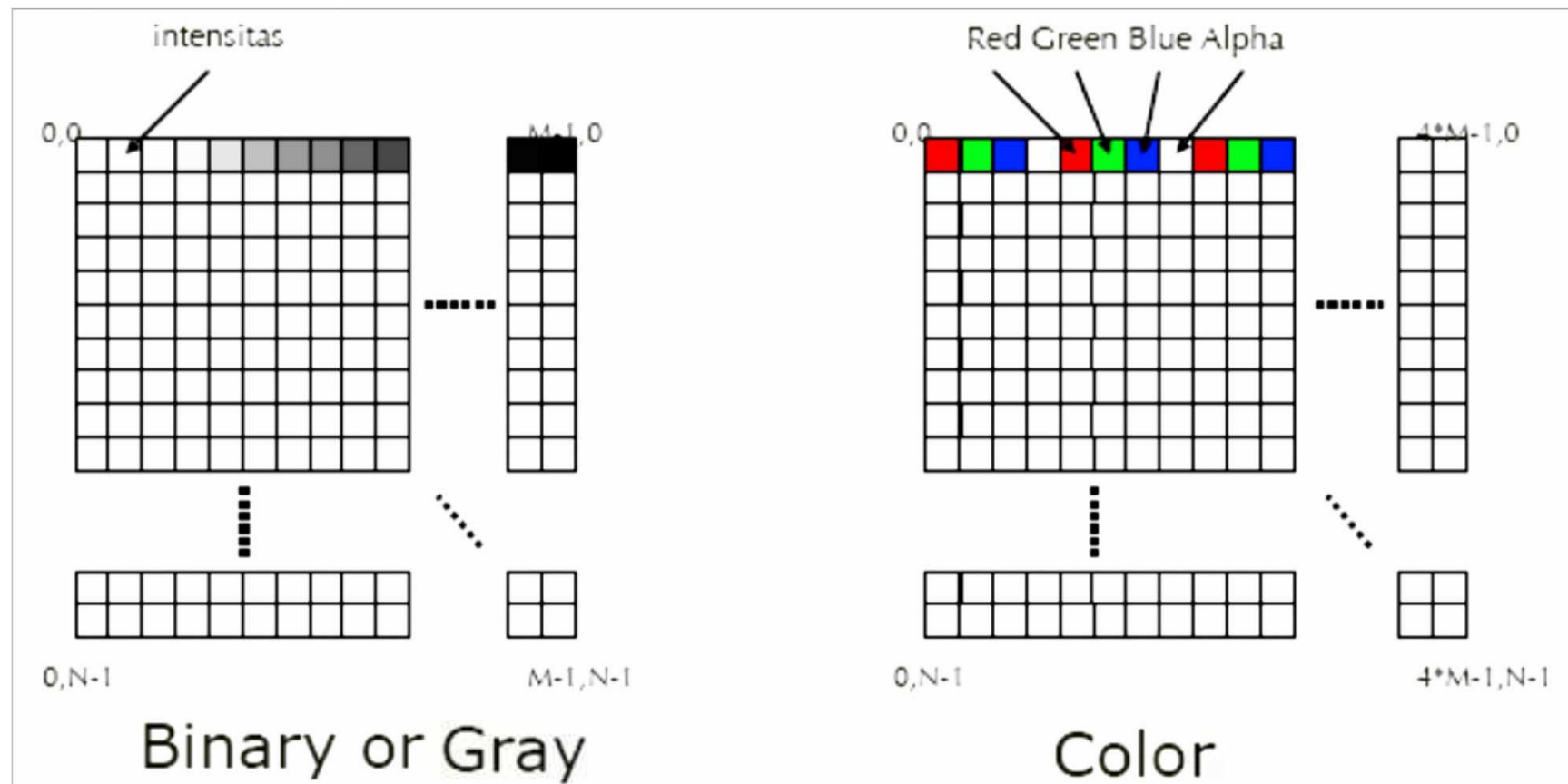


- Format Warna 24 BIT dinyatakan dengan :
11001001 01011001 00001011
R (8 bit) G (8 bit) B (8 bit)
- Masing-masing komponen warna RGB mempunyai nilai 0 s/d 255 (8bit) derajat kecerahan (derajat keabuan)

Format Warna

Warna		R	G	B
Merah		255	0	0
Hijau		0	255	0
Biru		0	0	255
Kuning		255	255	0
Magenta		255	0	255
Cyan		0	255	255
Putih		255	255	255
Hitam		0	0	0
Abu-abu		128	128	128

Representasi Citra



Konversi RGB ke Gray Scale



Setiap pixel mempunyai nilai red (r), green (g) dan blue (b) dengan nilai masing-masing 0-255



Setiap pixel mempunyai nilai derajat keabuan x dengan nilai 0-255

$$x = \frac{r + g + b}{3}$$

$$x = a_r \cdot r + a_g \cdot g + a_b \cdot b$$

dimana : $a_r + a_g + a_b = 1$

Konversi Gray Scale ke Biner



Setiap pixel mempunyai nilai derajat keabuan x dengan nilai 0-255

Setiap pixel mempunyai nilai warna x_{bw} dengan nilai 0 dan 1

$$x_{bw} = \begin{cases} 1 & \text{jika } x \geq 128 \\ 0 & \text{jika } x < 128 \end{cases}$$

$$x_{bw} = \begin{cases} 1 & \text{jika } x \geq x \\ 0 & \text{jika } x < x \end{cases}$$

Konversi Gray Scale ke m-Bit



Setiap pixel mempunyai nilai derajat keabuan x dengan nilai 0-255



Setiap pixel mempunyai nilai warna x_{th} dengan nilai 0 sampai dengan $2^m - 1$

$$x_{th} = (2^m) \text{int} \left(\frac{x}{2^m} \right)$$

Contoh :

$X=100$, gray scale 4 bit (0-64)

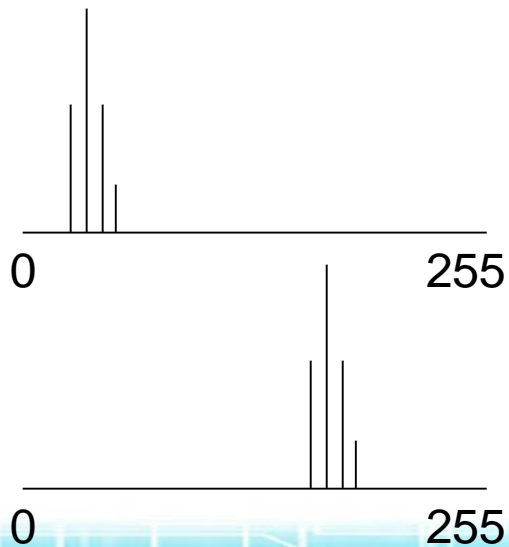
$X_{baru} = 64 \times (100/64)$

$= 64 \times 1 = 64$

Pengaturan Brightness



Proses pengaturan brightness adalah proses penambahan nilai derajat keabuan x dengan nilai perubahan brightness $t_{\text{brightness}}$



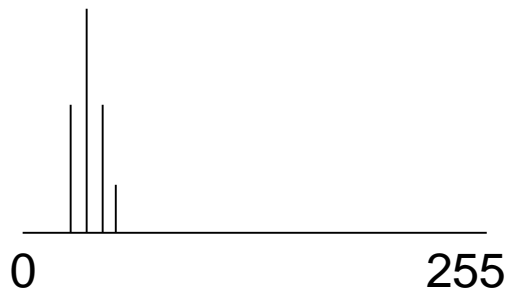
$$x_{\text{brightness}} = x + t_{\text{brightness}}$$

$t_{\text{brightness}}$ bisa positif dan negatif

Pengaturan Contrast

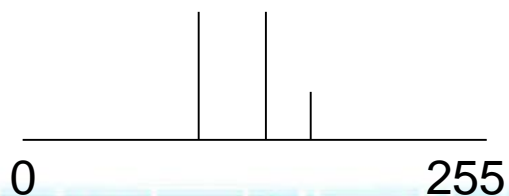


Proses pengaturan contrast adalah proses perkalian nilai derajat keabuan x dengan nilai perubahan contrast t_{contrast}



$$x_{\text{contrast}} = x \times t_{\text{contrast}}$$

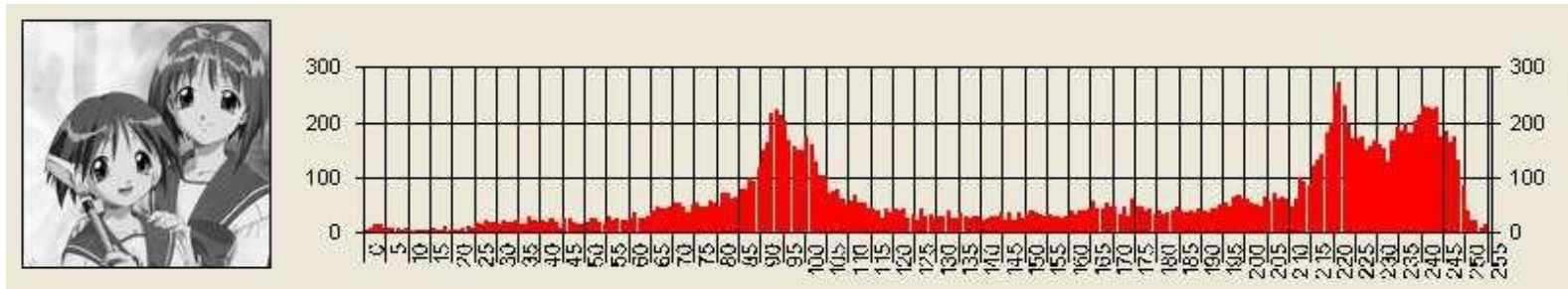
$0 < t_{\text{kontras}} < m$, dengan m positif



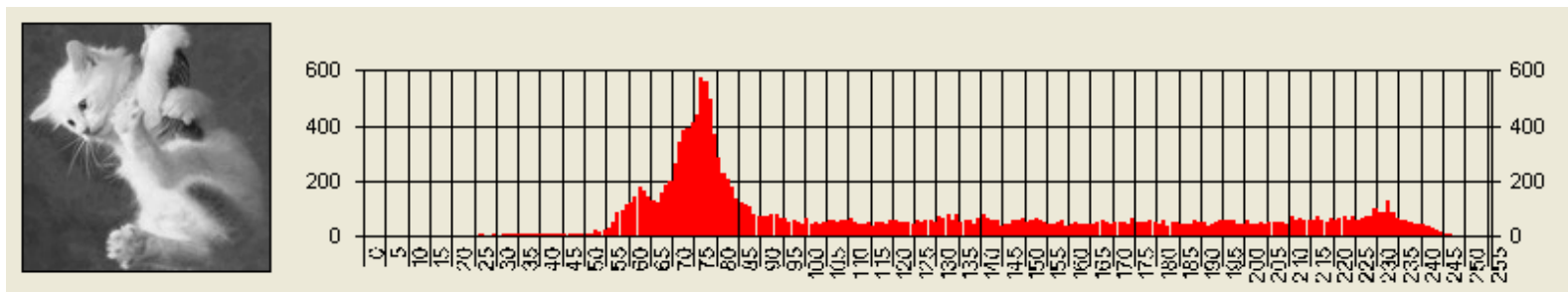
Gray Scale Histogram

- Histogram di dalam gambar gray-scale → **distribusi dari derajat keabuan (terang/gelap)** pada suatu gambar.
- Dari histogram ini dapat dilihat :
 - Gambar tersebut lebih banyak warna gelap atau
 - Lebih banyak warna terang
- **Histogram Equalization adalah** suatu teknik untuk meratakan distribusi terang/gelap sehingga gambar kelihatan lebih jelas
 - Teknik histogram ini dapat dikembangkan untuk memperbaiki kualitas gambar (image enhancement)

Gray Scale Histogram



Gambar ini didominasi warna terang, karena grafik di sebelah kanan terlihat lebih banyak.



Gambar ini didominasi warna gelap, karena grafik di sebelah kiriterlihat lebih banyak.

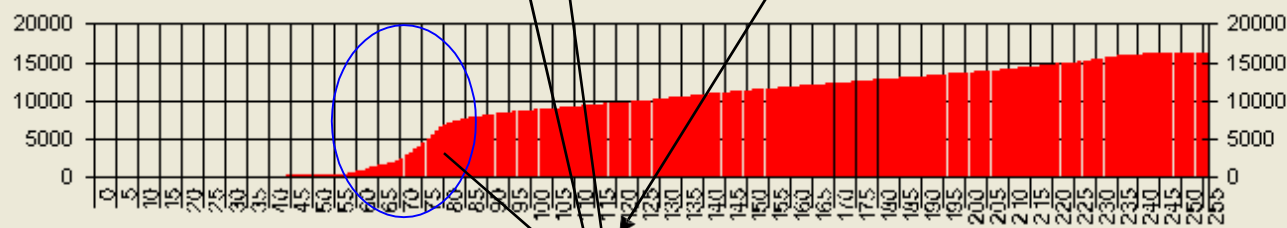
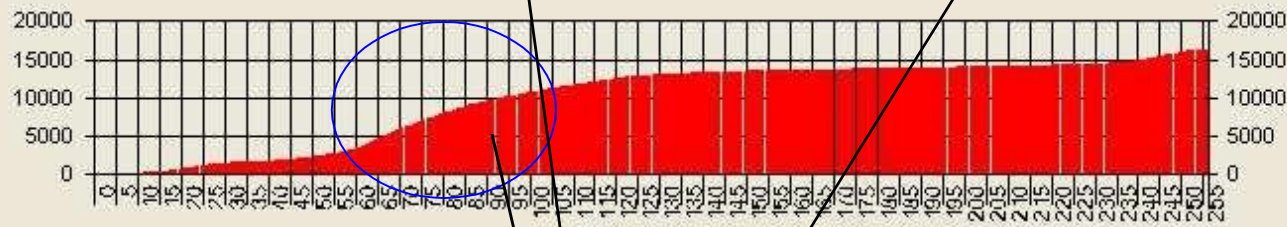
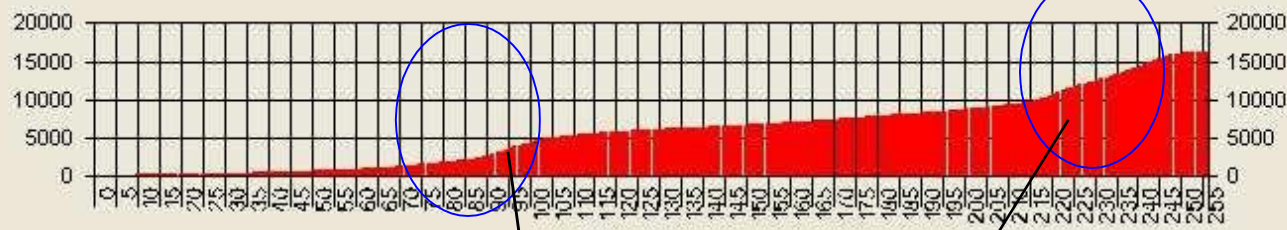
Distribusi Kumulatif

- **Distribusi kumulatif $C(x)$** adalah nilai total histogram dari tingkat keabuan=0 sampai dengan tingkat keabuan= x , dan didefinisikan dengan:

$$C(x) = \sum_{w=0}^x H(w)$$

- **Dapat digunakan** untuk menunjukkan perkembangan dari setiap step derajat keabuan.
- **Gambar dikatakan baik bila** mempunyai distribusi kumulatif yang pergerakannya hampir sama pada semua derajat keabuan.

Distribusi Kumulatif



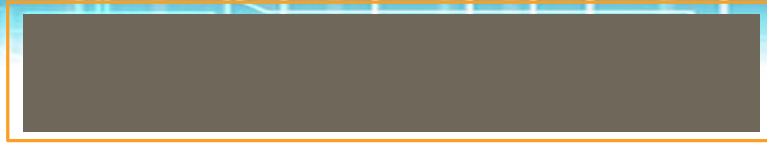
Perubahan yang tajam

Distribusi Kumulatif

- Gambar-gambar hasil photo mempunyai perubahan yang tidak terlalu tajam dan biasanya tidak lebih dari satu. Hal ini menunjukkan **tingkat gradiasi yang halus pada gambar hasil photo.**
- Gambar-gambar kartun mempunyai banyak perubahan yang tajam, hal ini menunjukkan **tingkat gradiasi pada gambar kartun rendah (kasar).**

Histogram Equalization

- **Histogram Equalization adalah** suatu proses untuk meratakan histogram agar derajat keabuan dari yang paling rendah (0) sampai dengan yang paling tinggi (255) mempunyai kemunculan yang rata.
- Dengan histogram equalization hasil gambar yang memiliki **histogram yang tidak merata** atau **distribusi kumulatif yang banyak loncatan gradiasinya** akan menjadi **gambar yang lebih jelas** karena derajat keabuannya tidak dominan gelap atau dominan terang.
- **Proses histogram equalization ini meng urai** **distribusi kumulatif**, karena dalam proses ini dilakukan perataan gradien dari distribusi kumulatifnya.



Formulasi Histogram Equalization

- Histogram Equalization dari suatu distribusi kumulatif C adalah:

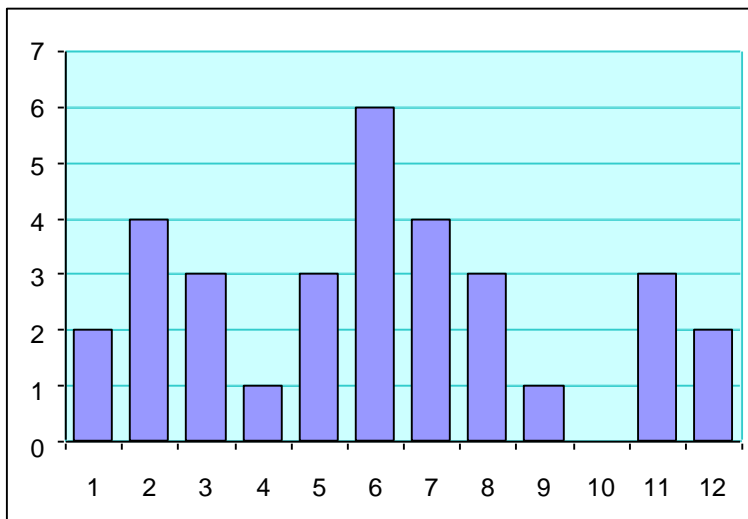
$$w = \frac{c_w \cdot t}{n_x \cdot n_y}$$

- Dengan :
 - C_w : nilai distribusi kumulatif pada derajat keabuan w
 - t : nilai threshold derajat keabuan = 2^8 atau 256
 - $n_x \cdot n_y$: ukuran gambar.

Perhitungan Histogram Equalization

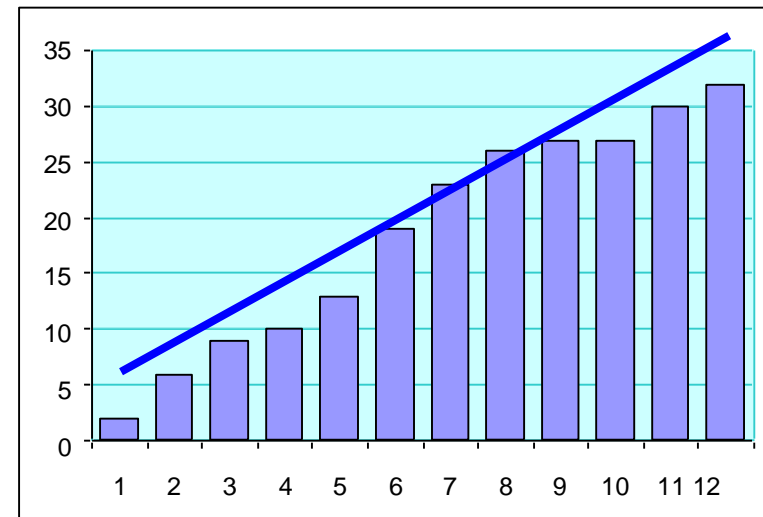
Perhatikan histogram berikut:

2 4 3 1 3 6 4 3 1 0 3 2



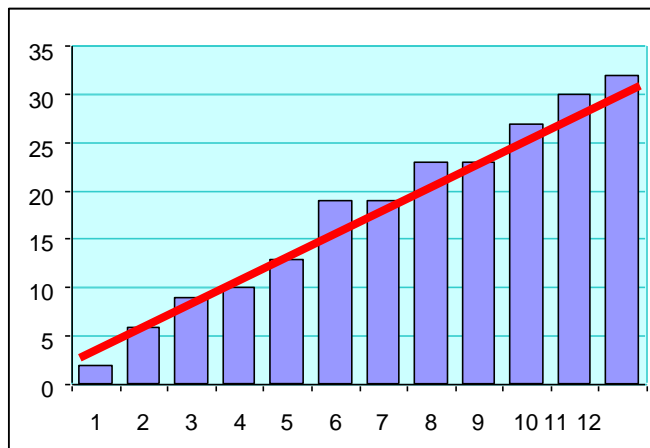
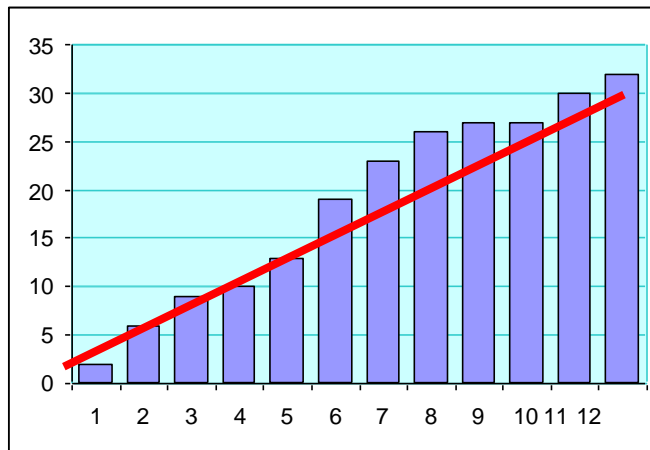
Distribusi Kumulatifnya

2 6 9 10 13 19 23 26 27 27 30 32



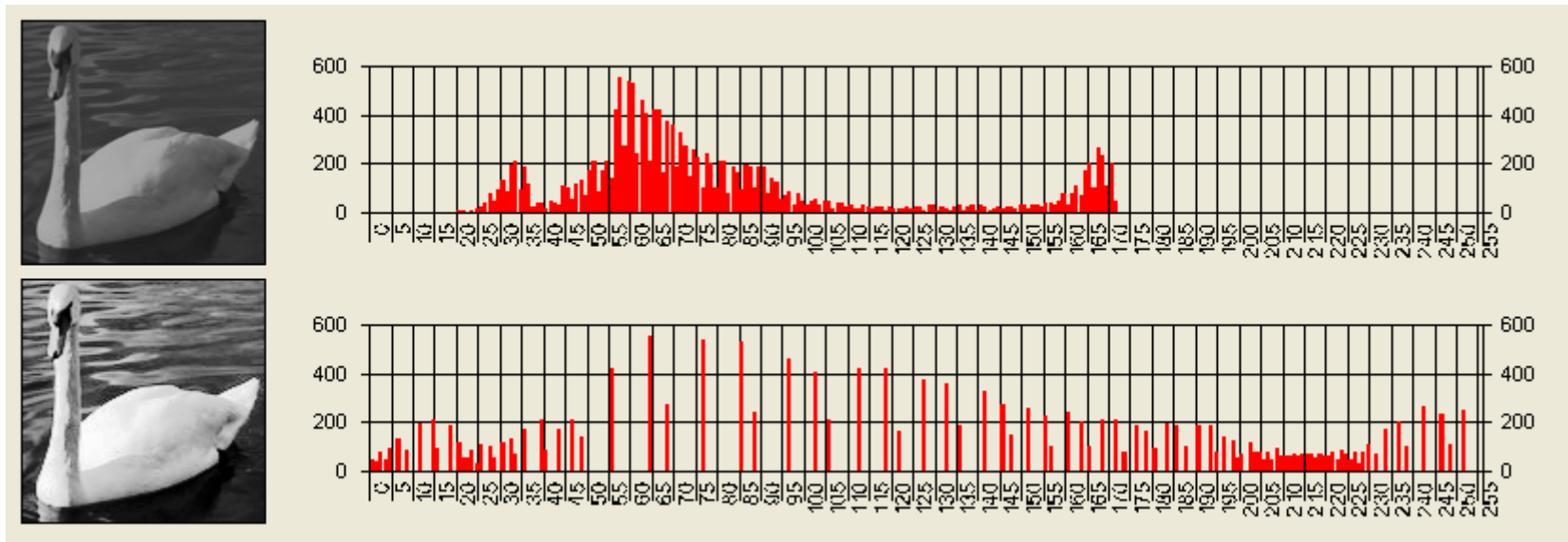
Perhitungan Histogram Equalization

Distribusi Kumulatif: 2 6 9 10 13 19 23 26 27 27 30 32

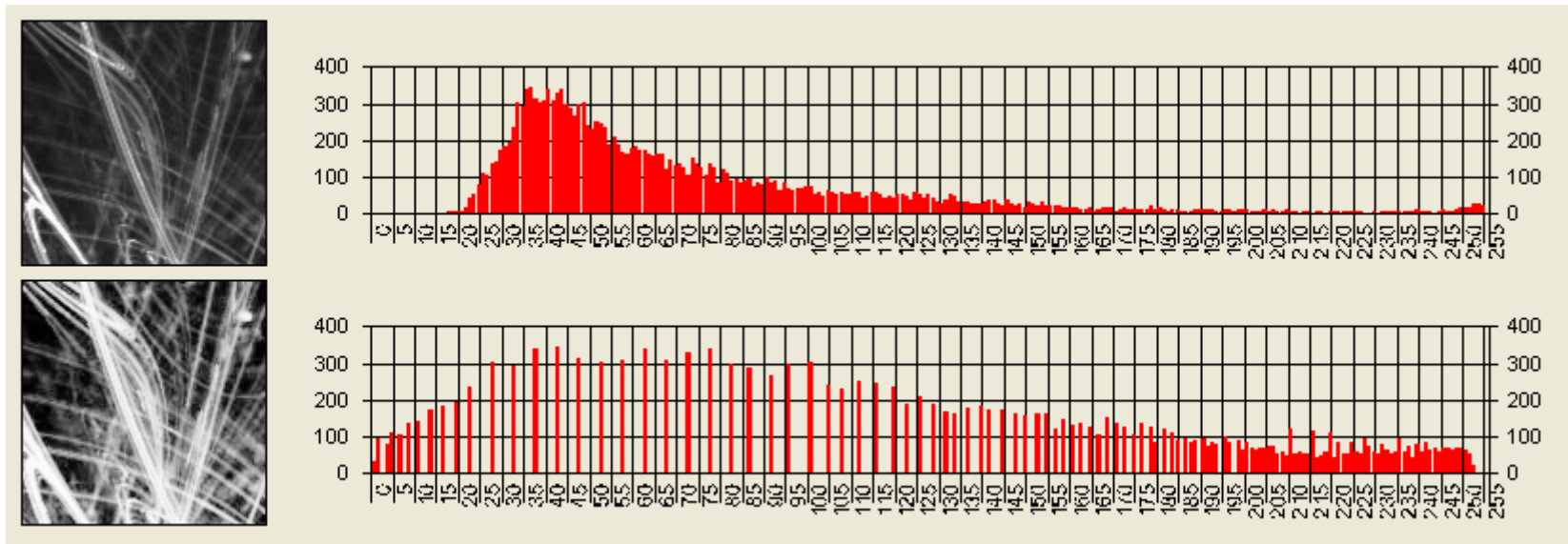


w	Cw	w-baru
		$(2 \cdot 12) / 41$
1	2	1
2	6	2
3	9	3
4	10	4
5	13	5
6	19	7
7	23	9
8	26	10
9	27	10
10	27	10
11	30	11
12	32	12

Histogram Equalization Pada Gambar



Histogram Equalization Pada Gambar



Histogram Equalization Pada Gambar

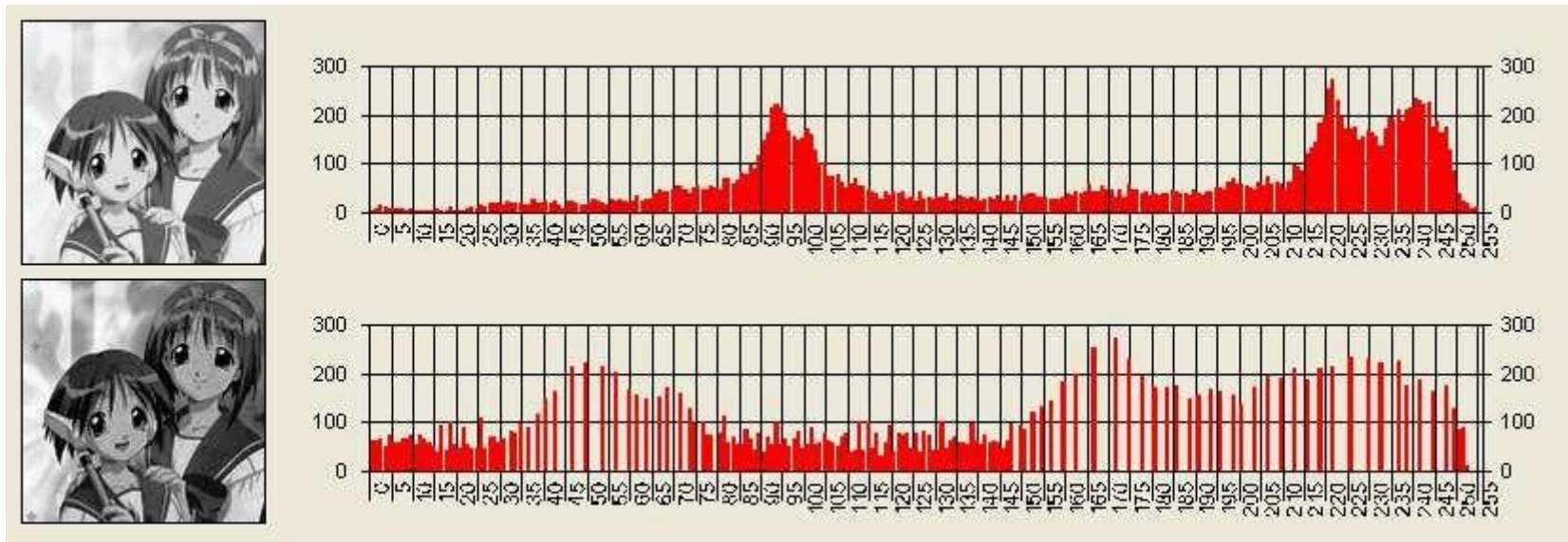
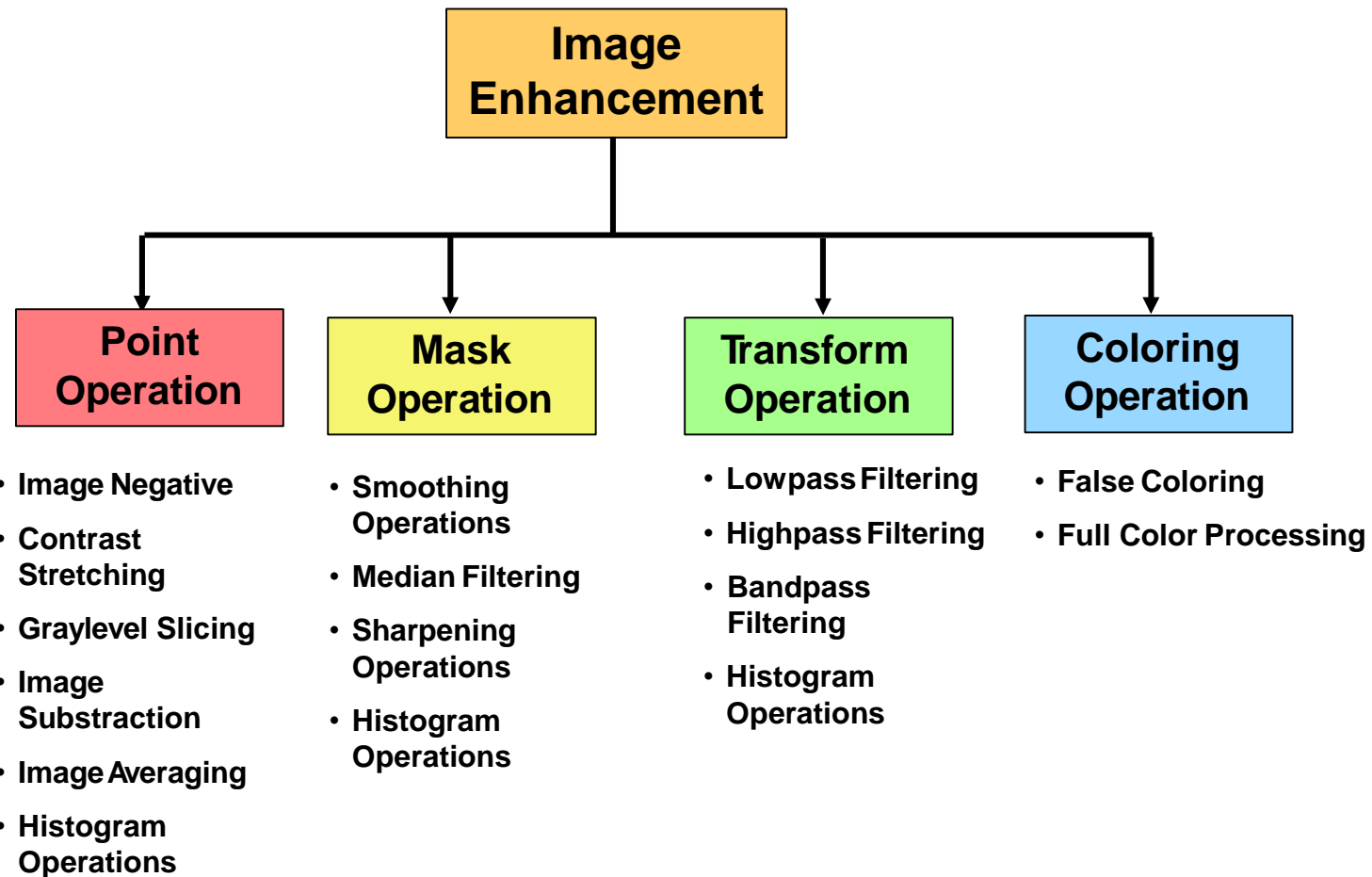


Image Enhancement

- **Image Enhancement adalah** proses agar citra menjadi lebih baik secara visual untuk aplikasi tertentu
- Proses sangat bergantung pada kebutuhan, dan pada keadaan citra input
- **Image Enhancement dapat dilakukan** dalam
 - **Spatial Domain** (dilakukan pada citra asli)
 - $g(m,n) = T[f(m,n)]$
 - **Frequency Domain** (dilakukan pada hasil DFT citra)
 - $G(u,v) = T[F(u,v)]$ dimana
 - $G(u,v) = F[g(m,n)]$ dan $F(u,v) = F[f(m,n)]$

Teknik Image Enhancement

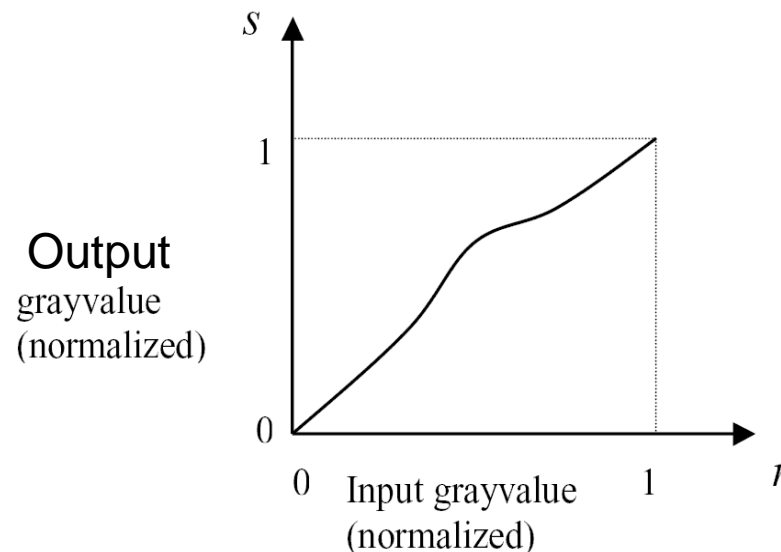


Point Operation

- Cara paling mudah untuk melakukan peningkatan mutu pada domain spasial
- Melibatkan satu piksel saja (tidak menggunakan jendela ketetanggaan)
- Pengolahan menggunakan histogram juga termasuk dalam bagian point processing

Point Operation

- Output pixel $g(m,n)$ hanya berdasar input sebuah pixel $f(m,n)$. Pixel tetangga tidak berpengaruh.
- Biasanya **dinotasikan** sebagai: **$s=T(r)$**
- Digambarkan seperti fungsi sbb:



Point Operation

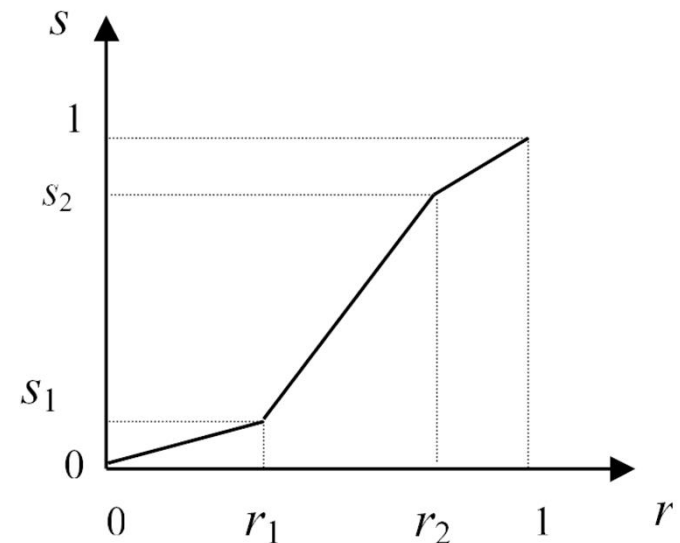
- **Image Negative**

- Mengubah nilai gray-level piksel citra input dengan $G_{baru} = 255 - G_{lama}$
- Hasilnya seperti klise foto



Point Operation

- **Contrast Stretching**
 - **Kontras adalah** tingkat penyebaran piksel-piksel ke dalam intensitas warna.
 - Mengubah kontras dari suatu image dengan cara mengubah grey level piksel-piksel pada citra menurut fungsi $s = T(r)$ tertentu
 - $r_1 \leq r_2, s_1 \leq s_2$
 - $r_1 = r_2, s_1 = s_2 \rightarrow$ tidak ada perubahan
 - $r_1 = r_2, s_1 = 0, s_2 = 255 \rightarrow$ thresholding menjadi citra biner dengan ambang r_1



Point Operation

□ Contrast Stretching

- Salah satu fungsi kontras secara matematis adalah sbb:

$$f_o(x, y) = G \cdot (f_i(x, y) - P) + P$$

□ Dengan :

- G : bilangan skalar, sebagai koefisien penguatan kontras
- P : matriks konstan yang dipakai sebagai pusat pengontrasan
- $F_o(x, y)$: matriks hasil kontras
- $F_i(x, y)$: matriks citra asal

Point Operation

□ Contrast Stretching

□ Contoh :

□ Matriks ukuran 5x5 piksel akan dilakukan kontras dengan koefisien penguatan kontras $G=2$ dan pusat pengontrasan $P=10$. Perhitungan fungsi kontras dilakukan sebagai berikut :

$$2. \left\{ \begin{bmatrix} 15 & 30 & 40 & 50 & 45 \\ 25 & 25 & 25 & 40 & 30 \\ 40 & 50 & 35 & 45 & 35 \\ 50 & 40 & 15 & 25 & 40 \\ 30 & 40 & 20 & 25 & 20 \end{bmatrix} - \begin{bmatrix} 10 & 10 & 10 & 10 & 10 \\ 10 & 10 & 10 & 10 & 10 \\ 10 & 10 & 10 & 10 & 10 \\ 10 & 10 & 10 & 10 & 10 \\ 10 & 10 & 10 & 10 & 10 \end{bmatrix} \right\} + \begin{bmatrix} 10 & 10 & 10 & 10 & 10 \\ 10 & 10 & 10 & 10 & 10 \\ 10 & 10 & 10 & 10 & 10 \\ 10 & 10 & 10 & 10 & 10 \\ 10 & 10 & 10 & 10 & 10 \end{bmatrix} = \begin{bmatrix} 20 & 50 & 70 & 90 & 80 \\ 40 & 40 & 40 & 70 & 50 \\ 70 & 90 & 60 & 80 & 60 \\ 90 & 70 & 20 & 40 & 70 \\ 50 & 70 & 30 & 40 & 30 \end{bmatrix}$$

MatriksAsal

Matriks Konstan

Matriks Konstan

Matriks Hasil

Point Operation

- **Contrast Stretching**
- **Ada tiga macam kontras:**
 - **Kontras Rendah**
 - Kontras ini terjadi karena kurangnya pencahayaan
 - Memiliki kurva histogram yang sempit (tepi paling kanan berdekatan dengan tepi paling kiri)
 - Artinya : titik tergelap dalam citra tidak mencapai hitam pekat dan titik paling terang tidak mencapai berwarna putih cemerlang
 - **Kontras Tinggi**
 - Memiliki kurva histogram yang terlalu lebar
 - Intensitas terang dan gelap merata ke seluruh skala intensitas
 - **Kontras Normal**
 - Lebar kurva histogram terlalu sempit dan tidak terlalu melebar

Point Operation

□ Contoh Contrast Stretching



Point Operation

Peregangan Kontras

- **Contoh Contrast Stretching**

Kontras rendah



Kontras tinggi



Kontras bagus



Point Operation

- **Thresholding (Ambang Batas)**
 - **Thresholding adalah** proses mengubah citra berderajat keabuan menjadi citra biner atau hitam putih, sehingga dapat diketahui daerah mana yang termasuk objek dan background dari citra secara jelas.
 - Citra hasil *thresholding* biasanya digunakan lebih lanjut untuk proses pengenalan objek serta ekstraksi fitur.
 - Fungsi threshold :

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

- Dengan T : nilai threshold yang diberikan

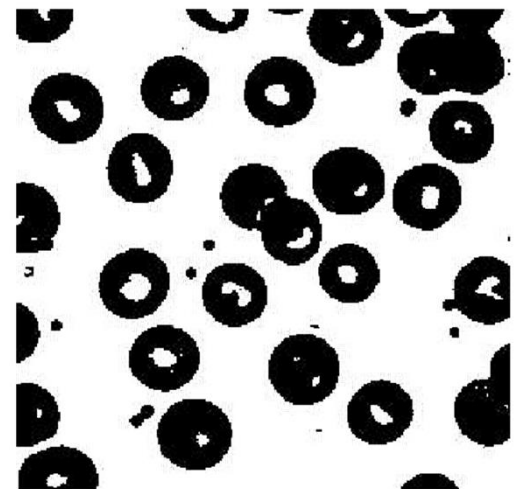
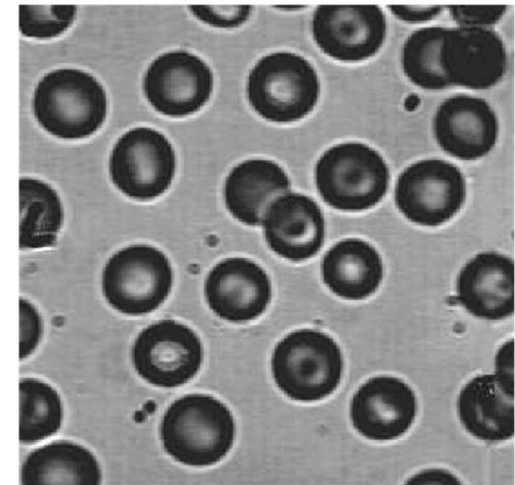
Point Operation

□ Contoh Thresholding

- Citra hitam putih yang memiliki *grayscale* 256, dipetakan menjadi citra biner (hanya mempunyai 2 warna, yaitu hitam dan putih), yang menggunakan fungsi transformasi sbb:

$$f_o(x,y) = \begin{cases} 0, & f_i(x,y) < 128 \\ 255, & f_i(x,y) \geq 128 \end{cases}$$

- Hasilnya adalah elemen-elemen matriks yang nilainya dibawah 128 diubah menjadi 0 (hitam), sedangkan elemen-elemen matriks yang nilainya diatas 128 diubah menjadi 255 (putih)



Point Operation

□ Contoh Perhitungan Thresholding

- Misalkan diketahui citra grayscale 256 warna dengan ukuran 5x5 piksel

40	160	69	170	123
20	250	140	80	90
70	30	128	115	85
40	234	70	211	125
20	34	80	221	30

- Maka perhitungan yang dilakukan adalah :
 - Setiap elemen-elemen matriks yang nilainya <128 diubah menjadi 0, sedangkan setiap elemen yang nilai ≥ 128 diubah menjadi 255