TEKNIK PENINGKATAN MUTU CITRA

Tujuan Perbaikan Citra

- Tujuan dari teknik peningkatan mutu citra adalah untuk melakukan pemrosesan terhadap citra agar hasilnya mempunyai kualitas relatif lebih baik dari citra awal untuk aplikasi tertentu.
- Kata baik disini tergantung pada jenis aplikasi dan problem yang dihadapi

Teknik peningkatan mutu citra

- Teknik peningkatan mutu citra dapat dibagi menjadi dua:
  - Peningkatan mutu citra pada domain spasial
    - Point Processing
    - Mask Processing
  - Peningkatan mutu citra pada domain frekuensi

I. Point Processing

- Cara paling mudah untuk melakukan peningkatan mutu pada domain spasial adalah dengan melakukan pemrosesan yang hanya melibatkan satu piksel saja (tidak menggunakan jendela ketetanggaan)

- Pengolahan menggunakan histogram juga termasuk dalam bagian point processing

Domain Spasial

- Prosedur yang secara langsung memanipulasi pixel.

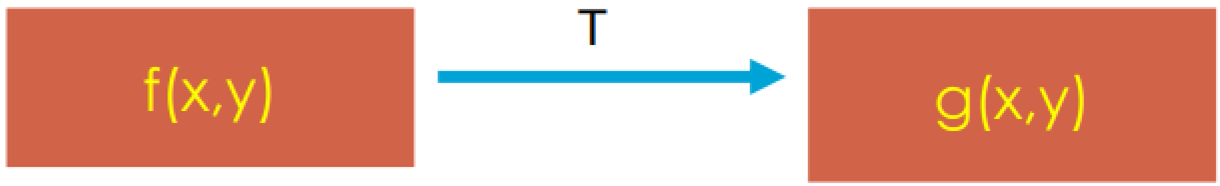
  $g(x,y) = T[f(x,y)]$

Dimana

$f(x,y)$ adalah image input

$g(x,y)$ adalah image yang diproses

T adalah sebuah operator pada f yang didefinisikan berdasar nilai neighborhood dari (x,y)



- Operator T dapat berupa :

Kumpulan pixels (x,y) dari image

Kumpulan dari 'neighborhoods' N(x,y) dari setiap pixel

Kumpulan dari images f1,f2,f3,...

Contoh Operator : div by 2

- Operasi terhadap himpunan pixel dari image

(Operator: Div. by 2)

Contoh Operator : sum

- Operasi terhadap kumpulan image f1,f2,…

6	8	2	0
12	200	20	10

5	5	1	0
2	20	3	4

(Operator: sum)

11	13	3	0
14	220	23	14

Point Processing

- Point processing, tetangga 1x1 piksel
  - Output pixel pada titik tertentu hanya bergantung pada input pixel pada titik tersebut dan tidak bergantung pada nilai pixel tetangganya
- g hanya bergantung pada nilai f pada posisi (x,y)
- T = fungsi transformasi gray level (atau intensitas mapping)

$$g(x,y) = T[f(x,y)]$$

II. Mask Processing

- Jika pada point processing kita hanya melakukan operasi terhadap masing-masing piksel, maka pada mask Processing kita melakukan operasi terhadap suatu jendela ketetanggaan pada citra.
- Kemudian kita menerapkan (mengkonvolusikan) suatu *mask* terhadap jendela tersebut. *Mask* sering juga disebut *filter.*

Teori Konvolusi

- Konvolusi terdapat pada operasi pengolahan citra yang mengalikan sebuah citra dengan sebuah mask atau kernel
- Operasi yang mendasar dalam pengolahan citra adalah operasi konvolusi

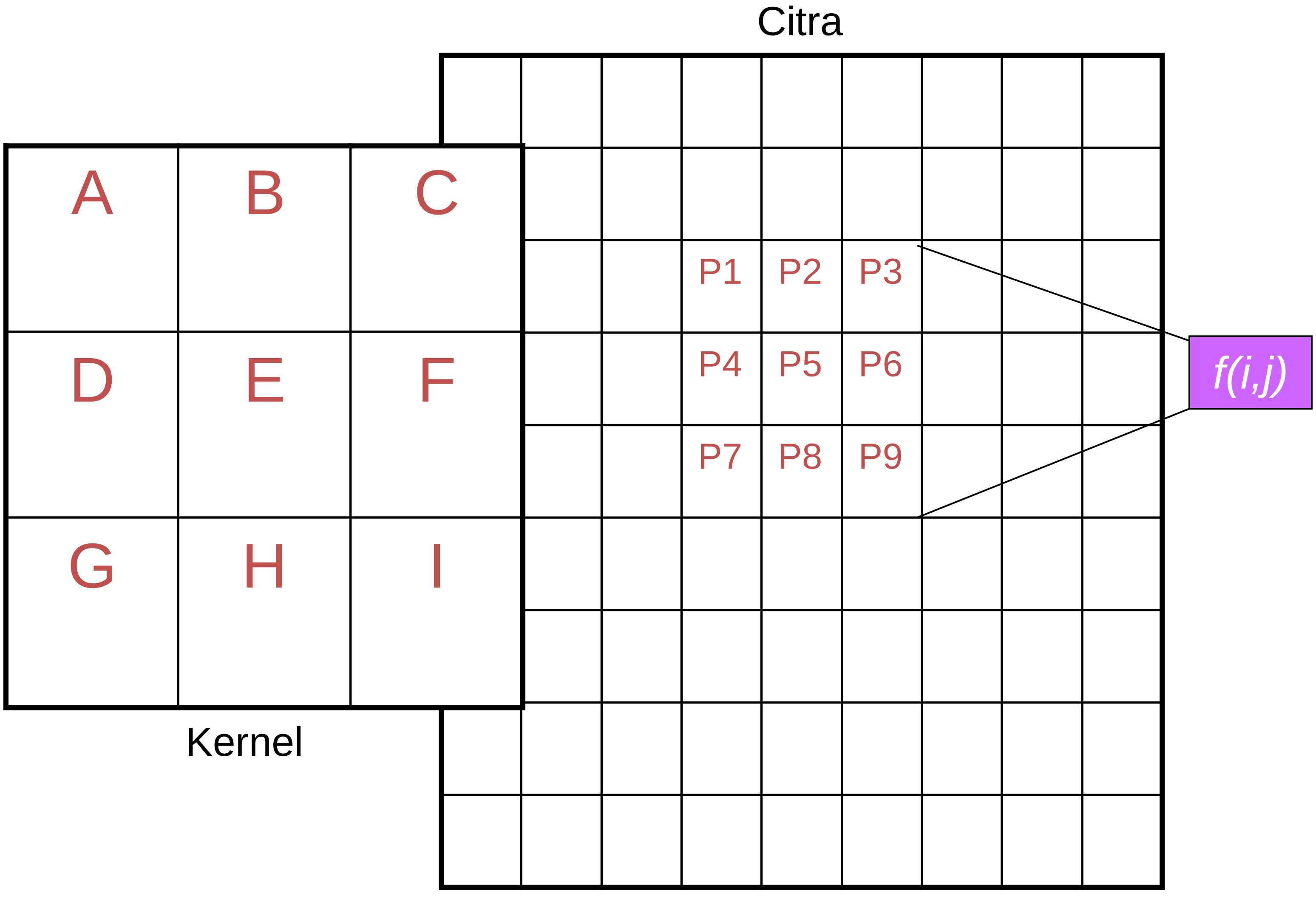
Teori Konvolusi

- Untuk fungsi diskrit, Konvolusi didefinisaikan sabagai :

$$h(x) = f(x) * g(x) = \int_{a=-\infty}^{\infty} f(a)\, g\,(x-a)$$

- Pada operasi konvolusi di atas, g(x) disebut kernel konvolusi atau filter
- Kernel g(x) merupakan suatu jendela yang dioperasikan secara bergeser pada sinyal masukan f(x), yang dalam hal ini, jumlah perkalian kedua fungsi setiap titik merupakan hasil konvolusi yang dinyatakan dengan keluaran h(x)

Ilustrasi Konvolusi



$$f(i,j) = AP1 + BP2 + CP3 + DP4 + EP5 + FP6 + GP7 + HP8 + IP9$$

Contoh Operasi Konvolusi

- Operasi konvolusi dilakukan dengan mengeser kernel konvolusi pixel per pixel
- Hasil konvolusi disimpan dalam matrik yang baru

$$
f(x,y) = \begin{pmatrix} 4 & 4 & 3 & 5 & 4 \\ 6 & 6 & 5 & 5 & 2 \\ 5 & 6 & 6 & 6 & 2 \\ 6 & 7 & 5 & 5 & 3 \\ 3 & 5 & 2 & 4 & 4 \end{pmatrix}
\qquad
g(x,y) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & \bullet 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}
$$

Citra 5 x 5

Kernel 3 x 3

Tanda ● menyatakan posisi (0,0) dari kernel

Contoh Operasi Konvolusi [1]

- Tempatkan kernel pada sudut kiri atas, kemudian hitung nilai pixel pada posisi (0,0) dari kernel

4	4	3	5	4
6	6	5	5	2
5	6	6	6	2
6	7	5	5	3
3	5	2	4	4

	3			

Hasil konvolusi =3. Nilai ini dihitung dengan cara berikut :

(0 x 4) + (-1 x 4) + (0 x 3) +(4 x 6)+ (-1 x 6) + (-1 x 5) + (0 x 5) + (-1 x 6) +(0 x 6) =3

Contoh Operasi Konvolusi [2]

- Geser kernel satu pixel ke kanan, kemudian hitung nilai pixel pada posisi **(0,0)** dari kernel

4	4	3	5	4
6	6	5	5	2
5	6	6	6	2
6	7	5	5	3
3	5	2	4	4

	3	0		

Hasil konvolusi =0. Nilai ini dihitung dengan cara berikut :

(0 x 4) + (-1 x 3) + (0 x 5) + (-1 x 6) + (4 x 5) + (-1 x 5) + (-1 x 6) +(0 x 6) + (-1 x 6) + (0 x 6) =0

Contoh Operasi Konvolusi [3]

- Geser kernel satu pixel ke kanan, kemudian hitung nilai pixel pada posisi (0,0) dari kernel

4	4	3	5	4
6	6	5	5	2
5	6	6	6	2
6	7	5	5	3
3	5	2	4	4

$\longrightarrow$

	3	0	2	

Hasil konvolusi =2. Nilai ini dihitung dengan cara berikut :

(0 x 3) + (-1 x 5) + (0 x 4) + (-1 x 5) + (4 x 5) + (-1 x 2) + (0 x 6) + (-1 x 6) + (0 x 2) = 2

Contoh Operasi Konvolusi [4]

- Selanjutnya, Geser *Kernel* Satu *Pixel* ke bawah, lalu mulai lagi melakukan Konvolusi dari sisi kiri citra. Setiap kali Konvolusi, Geser *Kernel* Satu *Pixel* Ke Kanan:

(i)

4	4	3	5	4
6	6	5	5	2
5	6	6	6	2
6	7	5	5	3
3	5	2	4	4

→

	3	0	2	
	0			

Hasil konvolusi =0. Nilai ini dihitung dengan cara berikut :

(0 x 6) + (-1 x 6) + (0 x 5) + (-1 x 5) + (4 x 6) + (-1 x 6) + (0 x 6) + (-1 x 7) + (0 x 5) = 0

Contoh Operasi Konvolusi [4]

(ii)

4	4	3	5	4
6	6	5	5	2
5	6	6	6	2
6	7	5	5	3
3	5	2	4	4

→

	3	0	2	
	0	2		

Hasil konvolusi =2. Nilai ini dihitung dengan cara berikut :

(0 x 6) + (-1 x 5) + (0 x 5) + (-1 x 6) + (4 x 6) + (-1 x 6) + (0 x 7) + (-1 x 5) + (0 x 5) =2

Contoh Operasi Konvolusi [4]

(iii)

4	4	3	5	4
6	6	5	5	2
5	6	6	6	2
6	7	5	5	3
3	5	2	4	4

	3	0	2	
	0	2	6	

Hasil konvolusi =6. Nilai ini dihitung dengan cara berikut :

(0 x 5) + (-1 x 5) + (0 x 2) + (-1 x 6) + (4 x 6) + (-1 x 2) + (0 x 5) + (-1 x 5) + (0 x 3) =6

Contoh Operasi Konvolusi [4]

- Dengan cara yang sama seperti tadi , maka *pixel – pixel* pada baris ke tiga dikonvolusi sehingga menghasilkan :

	3	0	2	
	0	2	6	
	6	0	2	

Sebagai catatan, Jika hasil Konvolusi menghasilkan nilai Pixel negatif, maka nilai tersebut di jadikan 0, sebaliknya jika hasil Konvolusi menghasilkan nilai *pixel* lebih besar dari nilai keabuan maksimum, maka nilai tersebut dijadikan nilai keabuan maksimum

Contoh Operasi Konvolusi [4]

Masalah timbul bila *Pixel* yang di konvolusi adalah *Pixel* pinggir (border), karena beberapa Koefisien Konvolusi tidak dapat di Posisikan pada *Pixel – pixel* Citra (“Efek Menggantung”), seperti contoh di bawah ini:

4	4	3	5	4	?
6	6	5	5	2	?
5	6	6	6	2	?
6	7	5	5	3	
3	5	2	4	4	

Masalah “Menggantung” Seperti ini Selalu Terjadi pada Pixel – pixel pinggir kiri, kanan, atas, dan bawah. Solusi untuk masalah ini adalah [SID95]:

Contoh Operasi Konvolusi [4]

1. *Pixel – pixel* pinggir di abaikan, tidak di – Konvolusi. Solusi ini banyak di pakai di dalam pustaka fungsi – fungsi pengolahan citra. Dengan cara seperti ini, maka pixel – pixel pinggir nilainya sama seperti citra asal. 2.
2. Dupliaksi elemen citra, misalnya elemen kolom pertama disalin ke kolom *M*+1, begitu juga sebaliknya, lalu konvolusi *pixel –pixel* pinggir tersebut.
3. Elemen yang di tandai dengan “?” diasumsikan bernilai 0 atau Konstanta yang lain, Sehingga pixel – pixel pinggir dapat di lakukan .

Contoh Operasi Konvolusi [4]

Solusi dengan ketiga pendekatan diatas mengasumsikan bagian pinggir Citra lebarnya sangat kecil (hanya satu pixel) relatif di bandingkan dengan ukuran citra sehingga pixel – pixel pinggir tidak memperlihatkan efek yang kasat mata.

4	4	3	5	4
6	3	0	2	2
5	0	2	6	2
6	6	0	2	3
3	5	2	4	4

Jenis – jenis filter spasial

- Smoothing filters:
  - Lowpass filter (linear filter, mengambil nilai rata-rata)
  - Median filter (non-linear filter, mengambil median dari setiap jendela ketetanggan)

- Sharpening filters:
  - Roberts, Prewitt, Sobel (edge detection)
  - Highpass filter

Lowpass filter

$$\begin{bmatrix} 8 & 8 & 8 & 8 \\ 8 & 17 & 8 & 8 \\ 8 & 8 & 8 & 8 \end{bmatrix}$$

(i) sebelum konvolusi

$$\begin{bmatrix} 8 & 8 & 8 & 8 \\ 8 & 9 & 9 & 8 \\ 8 & 8 & 8 & 8 \end{bmatrix}$$

(ii) setelah konvolusi

Nilai 9 ini diperoleh dari hasil perhitungan konvolusi:

$$f'(1,1) = (8 + 8 + 8 + 8 + 17 + 8 + 8 + 8 + 8)/9 = 81/9 = 9$$

Median filter

13	10	15	14	18
12	10	10	10	15
11	11	**35**	10	10
13	9	12	10	12
13	12	9	8	10

(a) *Pixel* bernilai 35 terkena derau

13	10	15	14	18
12	10	10	10	15
11	11	**10**	10	10
13	9	12	10	12
13	12	9	8	10

(b) 35 diganti dengan median dari kelompok 3 × 3 *pixel*

9 10 10 10 **10** 10 11 12 35

LATIHAN

Hitunglah nilai citra dibawah ini dengan

- **teknik konvolusi**
- **Filter median**
- **Filter lowpass**

$$
f(x,y) = \begin{pmatrix} 4 & 4 & 3 & 5 & 4 \\ 6 & 6 & 5 & 5 & 2 \\ 5 & 6 & 6 & 6 & 2 \\ 6 & 7 & 5 & 5 & 3 \\ 3 & 5 & 2 & 4 & 4 \end{pmatrix}
\qquad
g(x,y) = \begin{pmatrix} 0 & -1 & 0 \\ -2 & \bullet 3 & -2 \\ 0 & -1 & 0 \end{pmatrix}
$$

Citra 5 x 5

Kernel 3 x 3

Tanda ● menyatakan posisi (0,0) dari kernel