

OPERASI KETETANGGAN PIKSEL

PERTEMUAN KE 5

Agenda

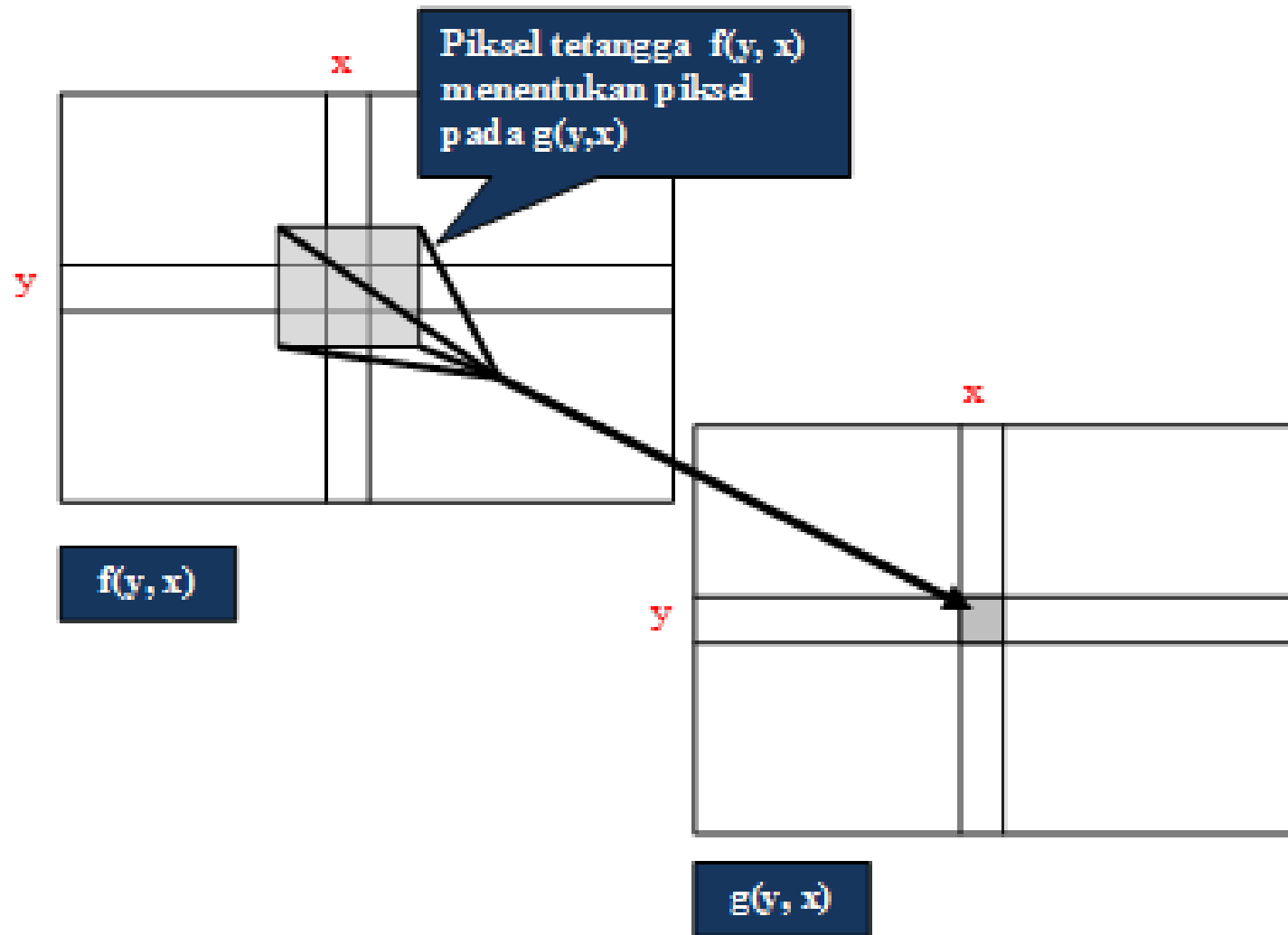
- Pengertian operasi ketetanggaan piksel
- Pengertian ketetanggaan piksel
- Aplikasi ketetanggaan piksel pada filter terbatas
- Pengertian konvolusi
- Problem pada konvolusi
- Mempercepat komputasi pada konvolusi



Operasi Ketetanggaan Piksel

Merupakan operasi pengolahan citra untuk mendapatkan nilai suatu piksel, yang melibatkan nilai piksel-piksel tetangganya.

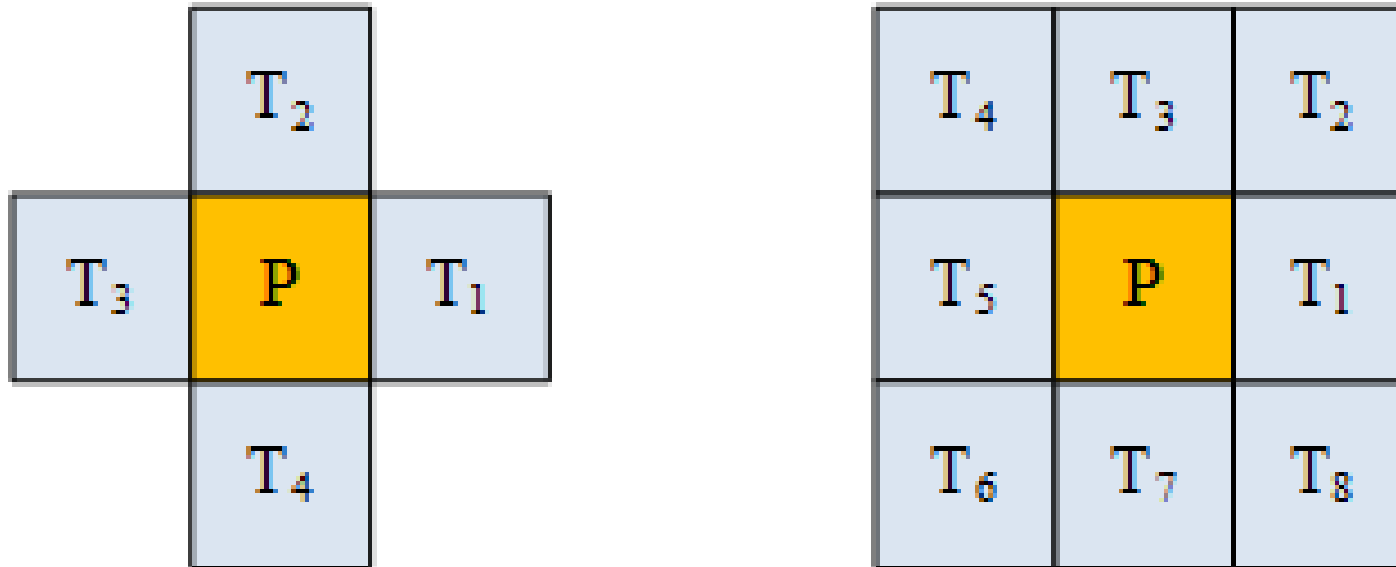
Hal ini didasarkan kenyataan bahwa setiap piksel pada umumnya tidak berdiri sendiri, melainkan terkait dengan piksel tetangganya, karena merupakan bagian suatu obyek tertentu didalam suatu citra.



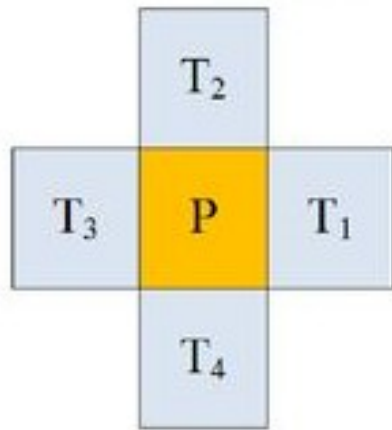
Gambar 5.1 Sebuah tetangga menentukan nilai sebuah piksel

Pengertian Ketetanggaan Piksel

Ketetanggaan piksel yang umum dipakai adalah 4 dan 8 ketetanggaan, seperti pada Gambar 5.2



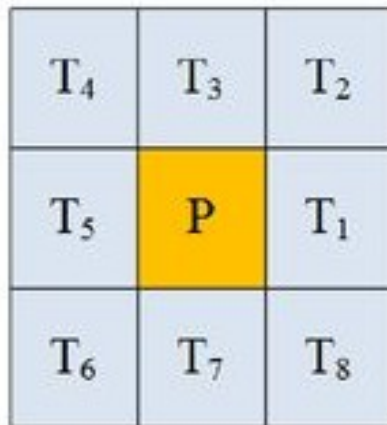
Gambar 5.2 Dua macam ketetanggaan piksel



$$T_1 = (b, k + 1), T_2 = (b - 1, k),$$

$$T_3 = (b, k - 1), T_4 = (b + 1, k)$$

...(1)



$$T_1 = (b, k + 1), T_2 = (b - 1, k - 1),$$

$$T_3 = (b, k - 1), T_4 = (b - 1, k - 1)$$

$$T_5 = (b, k - 1), T_6 = (b + 1, k - 1),$$

$$T_7 = (b + 1, k - 1), T_8 = (b + 1, k + 1)$$

...(2)

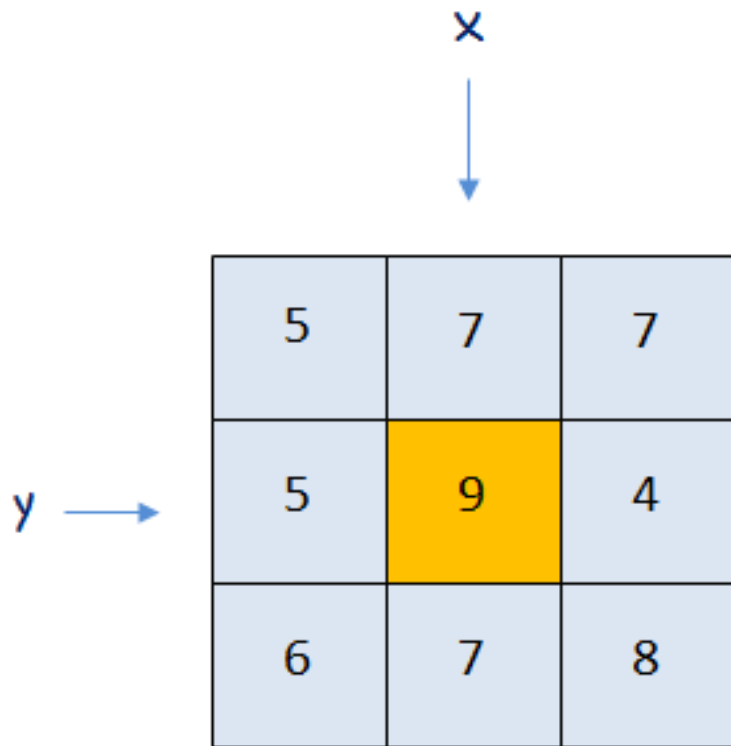
Gambar 5.3 Sebuah tetangga menentukan nilai sebuah piksel

Aplikasi Ketetanggaan Piksel pada Filter

- Filter Terbatas
- Filter Pererataan
- Filter Median

FILTER TERBATAS

Mencegah filter yang intensitasnya diluar intensitas piksel-piksel tetangga, seperti pada Gambar 5.4.



Berdasarkan keadaan disamping, maka untuk menerapkan filter terbatas, dapat diambil nilai minimum dan maksimumnya :

- $\text{minInt} = \text{minimum}(5, 7, 7, 5, 4, 6, 7, 8) = 4;$
- $\text{maksInt} = \text{maksimum}(5, 7, 7, 5, 4, 6, 7, 8) = 8;$
- **mengingat** $f(y, x)$ bernilai 9 dan **lebih besar dari pada 8** (maksInt) maka $g(y, x)$ **bernilai 8** ;
- **seandainya** $f(y, x)$ pada keadaan di atas **bernilai 2** (bukan 9), $g(y, x)$ akan **bernilai 4**.

Gambar 5.4 Contoh piksel dan tetangga

Listing program filter Batas :

```
% FILBATAS Melakukan operasi ketetanggaan piksel  
% menggunakan filter batas
```

```
F = imread('boneka2.tif');  
Ukuran = size(F);  
tinggi = Ukuran(1);  
lebar = Ukuran(2);  
  
G = F;  
for baris=2 : tinggi-1  
    for kolom=2 : lebar-1  
        minPiksel = min([F(baris-1, kolom-1) ...  
                        F(baris-1, kolom) F(baris, kolom+1) ...  
                        F(baris, kolom-1) ...  
                        F(baris, kolom+1) F(baris+1, kolom-1) ...  
                        F(baris+1, kolom) F(baris+1, kolom+1)]);  
        maksPiksel = max([F(baris-1, kolom-1) ...  
                          F(baris-1, kolom) F(baris, kolom+1) ...  
                          F(baris, kolom-1) ...  
                          F(baris, kolom+1) F(baris+1, kolom-1) ...  
                          F(baris+1, kolom) F(baris+1, kolom+1)]);  
  
        if F(baris, kolom) < minPiksel  
            G(baris, kolom) = minPiksel;  
        else  
            if F(baris, kolom) > maksPiksel  
                G(baris, kolom) = maksPiksel;  
            else  
                G(baris, kolom) = F(baris, kolom);  
            end  
        end  
    end  
end  
  
figure(1);  
imshow(G);  
  
clear;
```

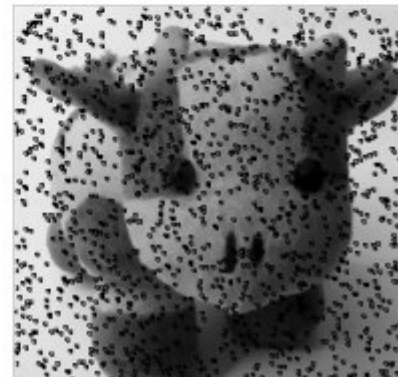
Hasil dari Penerapan filter Batas :



(a) Citra mobil yang telah diberi bintik-bintik putih



(b) Hasil pemfilteran gambar (a)



Gambar 5.5 Efek filter batas terhadap citra yang mengandung derau

FILTER PERERATAAN

Filter pererataan dilakukan dengan menggunakan rumus persamaan 3, Sebagai contoh piksel $f(y, x)$, dan delapan tetangga, memiliki nilai kecerahan seperti pada Gambar 5.4.

65	50	55
76	68	60
60	60	72

$$g(y, x) = \frac{1}{9} \sum_{p=-1}^1 \sum_{q=-1}^1 f(y + p, x + q) \dots\dots (3)$$

Gambar 5.6 Contoh piksel dan tetangga

FILTER PERERATAAN

65	50	55
76	68	60
60	60	72

Gambar 5.7 Contoh piksel dan tetangga

Pada contoh di samping, piksel dengan warna kuning (yaitu yang bernilai 68) merupakan nilai pada $f(y, x)$. Nilai rerata pengganti untuk $g(y, x)$ dihitung dengan cara seperti berikut:

$$g(y, x) = 1/9 \times (65+50+55+76+68+60+60+60+62) \\ = 61,7778$$

dibulatkan menjadi 62

Jadi, nilai 68 pada $f(y, x)$ diubah menjadi 62 pada $g(y, x)$.

Listing program filter pererataan :

```
% PEMERATAAN Melakukan operasi dengan filter pererataan

F = imread('boneka2.tif');
[tinggi, lebar] = size(F);

F2 = double(F);
for baris=2 : tinggi-1
    for kolom=2 : lebar-1
        jum = F2(baris-1, kolom-1)+ ...
              F2(baris-1, kolom) + ...
              F2(baris-1, kolom+1) + ...
              F2(baris, kolom-1) + ...
              F2(baris, kolom) + ...
              F2(baris, kolom+1) + ...
              F2(baris+1, kolom-1) + ...
              F2(baris+1, kolom) + ...
              F2(baris+1, kolom+1);

        G(baris, kolom) = uint8(1/9 * jum);
    end
end

figure(1); imshow(G);

clear;
```

Hasil dari Penerapan filter Pererataan :



(a) Citra mobil dengan bintik-bintik putih



(b) Hasil pemrosesan mobil



(c) Citra boneka berbintik dengan derau



(d) Hasil pemrosesan boneka

Gambar 5.8 Contoh penerapan filter pererataan

FILTER MEDIAN

Citra seperti pada Gambar 5.9. apabila di terapkan filter median, maka dilakukan dengan menggunakan rumus seperti persamaan 4.

10	13	10
10	10	12
12	12	12

Filter median sangat populer dalam pengolahan citra. Teorinya, filter ini sangat efektif dipakai untuk menghilangkan derau bintik-bintik.

Gambar 5.9 Contoh piksel dan tetangga

FILTER MEDIAN

10	13	10
10	10	12
12	12	12

10 10 10 10 12 12 12 12 13 ← Diurutkan

Nilai di tengah
(median)

Nilai yang lebih baik digunakan untuk suatu piksel ditentukan oleh nilai median dari setiap piksel, dan kedelapan piksel tetangga pada 8 ketetanggaan. Secara matematis, filter dapat dinotasikan seperti berikut:

$$g(y, x) = \text{median}(\begin{aligned} &f(y-1, x-1), f(y-1, x), f(y-1, x+1), \\ &f(y, x-1), f(y, x), f(y, x+1), \\ &f(y+1, x-1), f(y+1, x), f(y+1, x+1) \end{aligned}) \dots (4)$$

Gambar 5.10 Contoh piksel dan tetangga

Listing program filter Median :

% FILMEDIAN Melakukan operasi dengan filter median

```
F = imread('boneka2.tif');  
[tinggi, lebar] = size(F);
```

```
for baris=2 : tinggi-1  
    for kolom=2 : lebar-1  
        data = [F(baris-1, kolom-1) ...  
                F(baris-1, kolom) ...  
                F(baris-1, kolom+1) ...  
                F(baris, kolom-1) ...  
                F(baris, kolom) ...  
                F(baris, kolom+1) ...  
                F(baris+1, kolom-1) ...  
                F(baris+1, kolom) ...  
                F(baris+1, kolom+1)];
```

```
% Urutkan  
for i=1 : 8  
    for j=i+1 : 9  
        if data(i) > data(j)  
            tmp = data(i);  
            data(i) = data(j);  
            data(j) = tmp;  
        end  
    end  
end
```

```
% Ambil nilai median  
G(baris, kolom) = data(5);  
end  
end
```

```
figure(1); imshow(G);
```

```
clear;
```

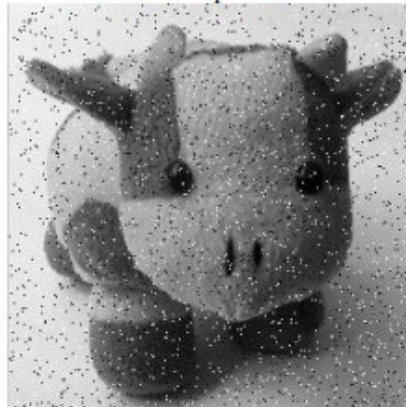
Hasil dari Penerapan filter Median :



(a) Citra mobil dengan bintik-bintik putih



(b) Hasil pemrosesan terhadap gambar (a)



(c) Citra boneka dengan derau



(d) Hasil pemrosesan terhadap gambar (c)

Gambar 5.11 Contoh penerapan filter median

TUGAS KE 2:

SILAHKAN ANDA COBA PENGGUNAAN DARI TEKNIK FILTER DIATAS, MENGGUNAKAN GAMBAR YANG BERBEDA. KEMUDIAN TENTUKAN FILTER MANA YANG TERBAIK MENURUT ANDA, KEMUDIAN JELASKAN ALASAN ANDA KENAPA FILTER TERSEBUT TERBAIK MENURUT ANDA.

Catatan : Pengumpulan tugas di SIP, sedangkan batas pengumpulan pada pertemuan minggu depan.

terimakasih
