

Tugas 3 Pengolahan Citra Video

Filter Spasial

Kenanya Keandra Adriel Prasetyo
5024211004

September 20, 2023

1 Pendahuluan

Pada Tugas 3 mata kuliah Pengolahan Citra Video, penulis diminta untuk membuat program untuk mengolah citra dari video dan melakukan berbagai macam filter seperti lowpass gaussian, highpass, gradient x, gradient y, band reject, dan bandpass.

2 Pembahasan Program

Kode keseluruhan pada Tugas 3 Pengolahan Citra Video ini dapat diakses pada https://github.com/KenanyaKAP/pcv_3

```
1 import numpy as np
2 import cv2
3
4 cap = cv2.VideoCapture(0)
5
6 # Generate a (2k+1)x(2k+1) gaussian kernel with mean=0 and sigma = s
7 s, k = 5, 10
8 probs = [np.exp(-z*z/(2*s*s))/np.sqrt(2*np.pi*s*s) for z in range(-k,k+1)
9          ]
9 kernelLow = np.outer(probs, probs)
10
11 # import matplotlib.pyplot as plt
12 # plt.imshow(kernelLow)
13 # plt.colorbar()
14 # plt.show()
15
16 kernelGradX = np.array([[1, -1]])
```

```

17 kernelGradY = np.array([[1, ],
18                        [-1,]])
19
20 while True:
21     ret, frame = cap.read()
22     img = np.double(cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY))/255
23
24     imgLow = cv2.filter2D(img, -1, kernelLow)
25     imgHigh = img - imgLow
26
27     imgGradX = cv2.filter2D(img, -1, kernelGradX)
28     imgGradY = cv2.filter2D(img, -1, kernelGradY)
29     imgGrad = np.sqrt(imgGradX**2 + imgGradY**2)
30
31     imgBandReject = imgLow + imgGrad
32     imgBandPass = img - imgBandReject
33
34     cv2.imshow('Citra Asli', img)
35     cv2.imshow('Low Pass', imgLow)
36     cv2.imshow('High Pass', imgHigh)
37
38     cv2.imshow('Gradient X', imgGradX*3)
39     cv2.imshow('Gradient Y', imgGradY*3)
40     cv2.imshow('Turunan Pertama', imgGrad*3)
41
42     cv2.imshow('Band Reject', imgBandReject)
43     cv2.imshow('Band Pass', imgBandPass*3)
44
45     if cv2.waitKey(1) == 27:
46         break
47
48 cap.release()
49 cv2.destroyAllWindows()

```

Program di atas akan mengambil gambar dari kamera secara realtime dan akan diolah secara langsung oleh program. Program diatas menerapkan beberapa filter, yakni LowPass Filter, HighPass Filter, GradientX Filter, GradientY Filter, Filter turunan pertama, Band Reject filter, dan Band Pass Filter.

2.0.1 LowPass Filter

```

1 s, k = 5, 10
2 probs = [np.exp(-z*z/(2*s*s))/np.sqrt(2*np.pi*s*s) for z in range(-k,k+1)
3         ]
4
5 kernelLow = np.outer(probs, probs)
6
7 imgLow = cv2.filter2D(img, -1, kernelLow)
8
9 cv2.imshow('Low Pass', imgLow)

```

Kode diatas digunakan untuk membuat kernel LowPass Filter. Kemudian Kernel tersebut digunakan ke dalam `cv2.filter2D` untuk melakukan Filter. Berikut adalah hasil citra asli dan setelah dilakukan filter. Fungsi LowPass yang digunakan adalah Gaussian Blur.



Figure 1: Citra Asli

2.0.2 HighPass Filter

```
1 imgHigh = img-imgLow
2
3 cv2.imshow('High Pass', imgHigh)
```

Sedangkan untuk melakukan HighPass Filter, kita bisa mendapatkannya dari Low-Pass Filter. Caranya adalah dengan mengurangi citra asli dengan LowPass, sehingga citra yang didapatkan tersisa sinyal High.

2.0.3 Gradient X

```
1 kernelGradX = np.array([[1, -1]])
2
3 imgGradX = cv2.filter2D(img, -1, kernelGradX)
4
5 cv2.imshow('Gradient X', imgGradX*3)
```



Figure 2: LowPass Filter

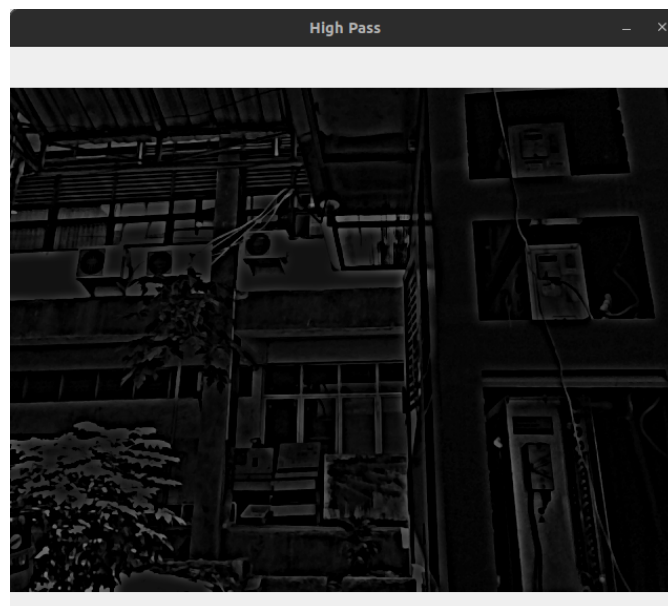


Figure 3: Citra Asli

Filter gradient X memiliki kernel yakni $[1, -1]$, filter ini akan mendeteksi sinyal high frekuensi pada axis X, sehingga gambar yang dihasilkan akan menampilkan garis vertical.



Figure 4: Gradient X

2.0.4 Gradient Y

```
1 kernelGradY = np.array([[1, ],
2                           [-1,]])
3
4 imgGradY = cv2.filter2D(img, -1, kernelGradY)
5
6 cv2.imshow('Gradient Y', imgGradY*3)
```

Filter gradient Y kebalikan dari gradient X. Filter ini memiliki kernel yakni $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$, filter ini akan mendeteksi sinyal high frekuensi pada axis Y, sehingga gambar yang dihasilkan akan menampilkan garis horizontal.

2.0.5 Turunan Pertama

```
1 imgGrad = np.sqrt(imgGradX**2 + imgGradY**2)
2
3 cv2.imshow('Turunan Pertama', imgGrad*3)
```

Sedangkan filter turunan pertama bisa didapatkan dari menggabungkan gradient x dan y. Caranya dengan memangkatkan gradient x dan y, kemudian ditambahkan dan di akar. Sehingga hasilnya adalah gabung dari gradient x dan y. Filter turunan pertama ini juga merupakan high filter.

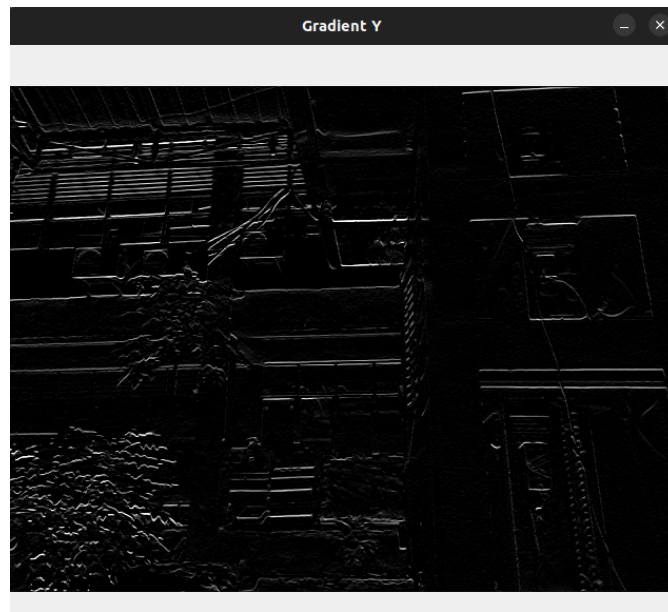


Figure 5: Gradient X

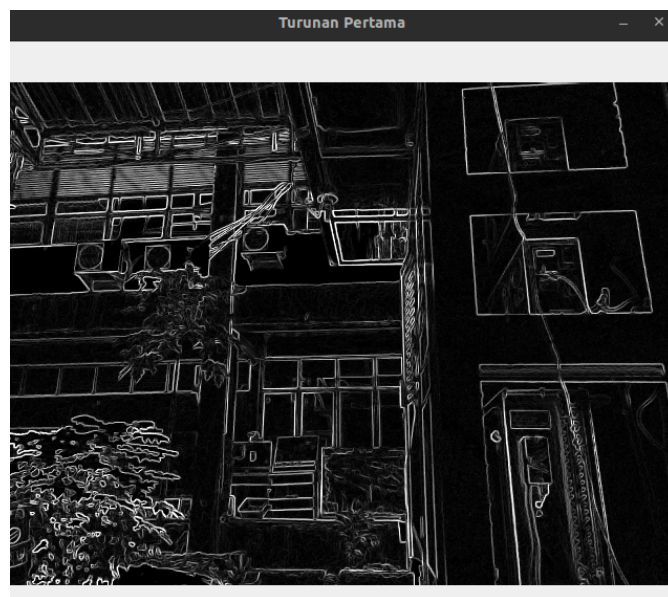


Figure 6: Turunan Pertama

2.0.6 Band Reject Filter

```
1 imgBandReject = imgLow + imgGrad
2
3 cv2.imshow('Band Reject', imgBandReject)
```

Filter Band Reject adalah gabungan dari Filter LowPass dan HighPass. LowPass filter yang digunakan adalah Gaussian, sedangkan HighPass Filter yang digunakan dari Filter turunan. Dengan menambahkan LowPass dan HighPass akan didapatkan Band Reject. berikut adalah gambar dari Band Reject.

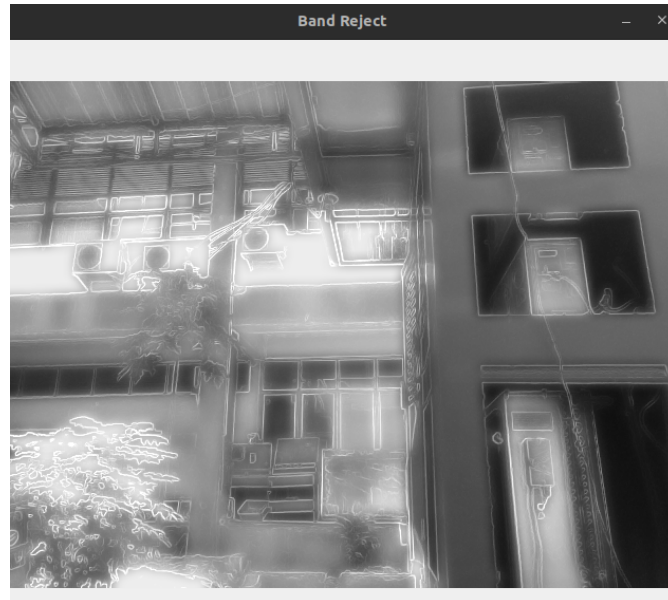


Figure 7: Band Reject Filter

2.0.7 Band Pass Filter

```
1 imgBandPass = img - imgBandReject
2
3 cv2.imshow('Band Pass', imgBandPass*3)
```

Filter Band Pass adalah kebalikan dari Band Reject Filter. Sehingga bisa didapatkan dengan mudah yakni mengurangi citra asli dengan Band Reject Filter. Sehingga didapatkanlah Band Pass Filter,

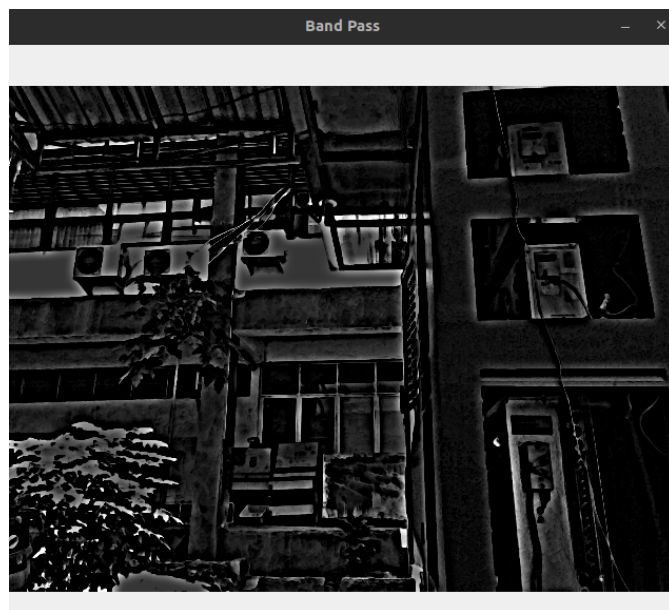


Figure 8: Band Pass Filter