Nama : Indah Hamidah

NIM : G64140055

Nama Dosen : Mayanda Mega Santoni

Nama Asisten : Kak Hana & Bang Tio

Lakukan sobel filter pada citra lena dengan nilai untuk parameter x\_order =1, y\_order=0, KernelSize=3,5 dan 7. Serta, lakukan sobel filter pada citra lena dengan nilai untuk parameter (x\_order=0, y\_order=1, KernelSize=3,5 dan 7). Bandingkan dan analisis hasil yang diberikan.

|  |
| --- |
| #include <iostream>  #include <cv.h>  #include <highgui.h>  using namespace std;  using namespace cv;  int main()  {  Mat src, src\_gray;  Mat dst;  Mat grad;  src = imread("cameraman.jpg");  dst.create(src.size(),src.type());  GaussianBlur( src, src, Size(3,3), 0, 0, BORDER\_DEFAULT );  cvtColor(src,src\_gray,CV\_BGR2GRAY);  Sobel( src\_gray, grad, CV\_16S, 1, 0, 7, 1, 0, BORDER\_DEFAULT );  convertScaleAbs(grad,dst);  imshow("sobel 7, x=1, y=0",dst);  waitKey(0);  } |

**Parameter x\_order =1, y\_order=0**

1. **Ukuran kernel 3**  b. **ukuran kernel 5**

|  |  |
| --- | --- |
| Sobel( src\_gray, grad, CV\_16S, 1, 0, 3, 1, 0, BORDER\_DEFAULT ); | Sobel( src\_gray, grad, CV\_16S, 1, 0, 5, 1, 0, BORDER\_DEFAULT ); |

1. Ukuran kernel 7

|  |
| --- |
| Sobel( src\_gray, grad, CV\_16S, 1, 0, 7, 1, 0, BORDER\_DEFAULT ); |

**Parameter x\_order =1, y\_order=0**

Ukuran kernel 3 Ukuran Kernel 5

|  |  |
| --- | --- |
| Sobel( src\_gray, grad, CV\_16S, 0,1, 3, 1, 0, BORDER\_DEFAULT ); | Sobel( src\_gray, grad, CV\_16S,0,1,5, 1, 0, BORDER\_DEFAULT ); |

Ukuran kernel 7

|  |
| --- |
| Sobel( src\_gray, grad, CV\_16S, 0, 1, 7, 1, 0, BORDER\_DEFAULT ); |

**Penjelasan:**

Untuk mendeteksi *edge* dengan metode Sobel, sebelumnya image dilakukan bluring terlebih dahulu, dimana fungsinya untuk menghilangkan noise-noise kecil seperti pada rumput-rumput pada *image cameraman.jpg*, sehingga rumpu tersebut tidak terdeteksi karena memiliki nilai *edge* yang kecil dibanding yang lainnya pada *image* tersebut.

Perbedaan parameter antara x=1, y=0 dan x=0, y=1 ialah dimana pada order\_x=0 dan order\_y=1 menghailkan garis tepian lebih mengarah ke tepian horizontal (kiri kanan) sedangkan pada parameter order\_x=1 dan order\_y=0 menghasilkan garis tepian lebih mengarah ke garis vertical (atas bawah).

Untuk KernelSize= 3,5, dan 7, semakin besar atau tinggi ukuran kernelnya maka akan semakin teliti dalam mendeteksi *edge* nya, dan dia dapat mendeteksi banyak *edge*. Jika *edge* nya terdeteksi maka akan diputihkan, jadi semakin banyak edge yang terdeteksi maka makin putih. Sehingga semakin teliti semakin putih karena banyak yang terdeteksi.