NHL STENDEN
computer vision & data science

Computer Vision

Linear filters (Convolution)

10 April 2018

j.van.de.loosdrecht@nhl.nl, jaap@vdlmv.nl

Linear filters

Overview:

- **Convolution algorithm**
- **Low pass filter**
- **High pass filter**
- **Sharpening**
- **Introduction to edge detection**

Convolution

Convolution (src, dest, divideFactor, edge, mask)

The convolution operator initialises a destination image by sliding a mask across a source image.

The pixel values under the mask are multiplied by the corresponding maskvalue. All products are summed and the sum is divided by a division factor.

This new value is assigned to the destination image at the position of the centre (= origin) of the mask.

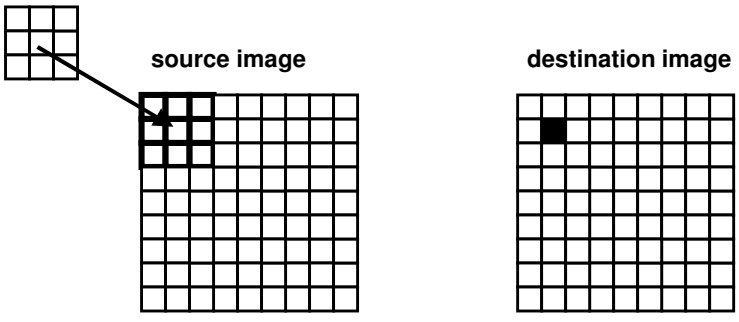
Convolution is a linear filter.

27-aug-18 Linear filters 3

Example with 3x3 mask and origin in centre

First pixel, first row

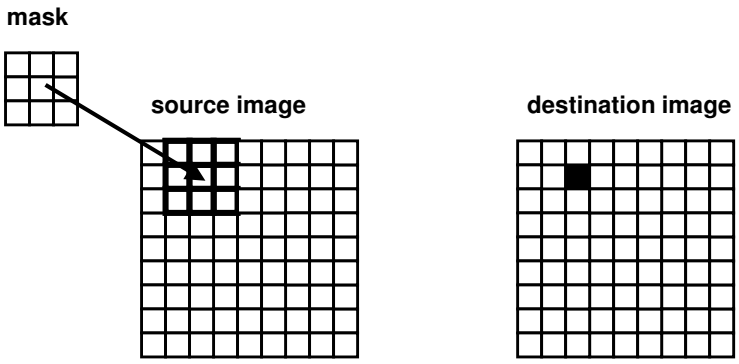
mask



27-aug-18 Linear filters 4

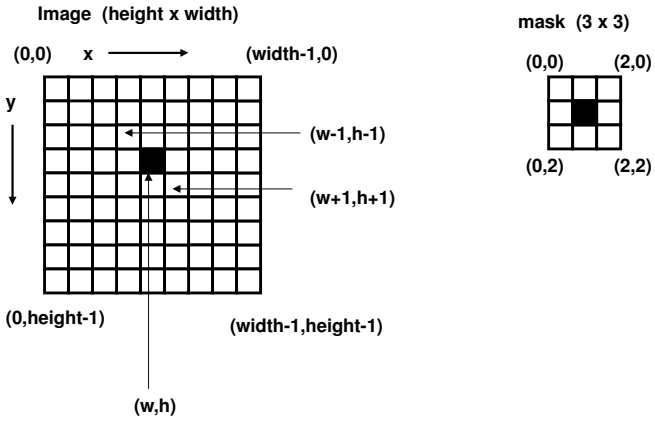
Example with 3x3 mask and origin in centre

Second pixel value, first row



27-aug-18 Linear filters 5

Convolution Algorithm



27-aug-18 Linear filters 6

Convolution Algorithm for 3x3 mask

Convolution (src, dest, divideFactor, edge, mask)

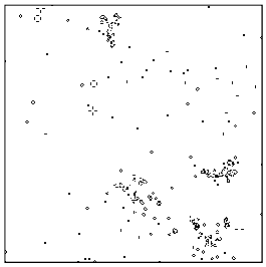
```
for (h = 1; h < height-1; h++) {
    for (w = 1; w < width-1; w++) {
        dest(w,h) = (mask(0,0) * src(w-1,h-1) +
                     mask(0,1) * src(w,h-1) +
                     mask(0,2) * src(w+1,h-1) +
                     mask(1,0) * src(w-1,h) +
                     mask(1,1) * src(w,h) +
                     mask(1,2) * src(w+1,h) +
                     mask(2,0) * src(w-1,h+1) +
                     mask(2,1) * src(w,h+1) +
                     mask(2,2) * src(w+1,h+1) ) / divideFactor;
    } // for w
} // for h
```


Example with 3x3 mask and origin in centre

What to do with not-initialised border pixels in destination image?

- **copy from source image**
- **set to zero**
- **remove from image**
- **extend border of image prior to operation**

Exercise John Conway's Game of Life (*)



Rules:

1. **Any live cell with fewer than two live neighbours dies, as if by loneliness.**
2. **Any live cell with more than three live neighbours dies, as if by overcrowding.**
3. **Any live cell with two or three live neighbours lives, unchanged, to the next generation.**
4. **Any dead cell with exactly three live neighbours comes to life.**

27-aug-18 Linear filters 9

Exercise John Conway's Game of Life (*)

Write script using framework:

```
Create gol ByteImage 256 256
Noise gol 6 1 1
SetLut gol Binary
while true
  Convolution gol gol2 1 EdgeExtend 3 3 1 1 1 1 1 1 1 10 1 1 1 1
  ThresholdMulti gol2 …….
  Display gol2
  Copy gol2 gol
endwhile
```

See gol_framework.jls for framework
See gol.jls for answer

27-aug-18 Linear filters 10

Low pass filter

Mask (in general):

- **mask values >= 0**
- **division factor = sum mask values**

Examples:

- **smooth mask:**

```
1  1  1
1  1  1
1  1  1
```

- **gaussian 5x5 mask:**

```
1   2   3   2   1
2   7  11   7   2
3  11  17  11   3
2   7  11   7   2
1   2   3   2   1
```

27-aug-18 Linear filters 11

Low pass filter

Usage:

- **removal of high frequencies**
- **noise removal**

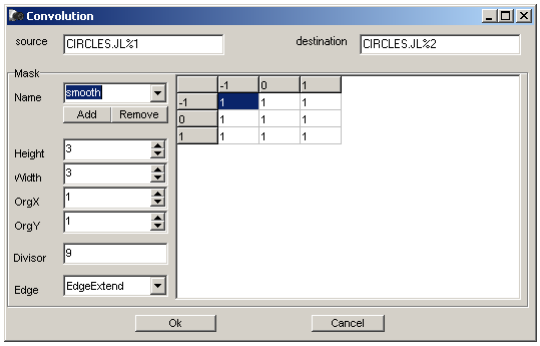
27-aug-18 Linear filters 12

Demonstration low pass filter

- Open image circles.jl
- Convolution with smooth mask use EdgeExtend
- Analyse pixels on circles.jl and convolution result
  - explain spatial frequency
  - smooth is low pass filter, it eliminates the high frequencies
- Convolution with gaussian 5x5 is a stronger low pass filter (no slides)
- Demo use of own defined masks and parameters (no slides)

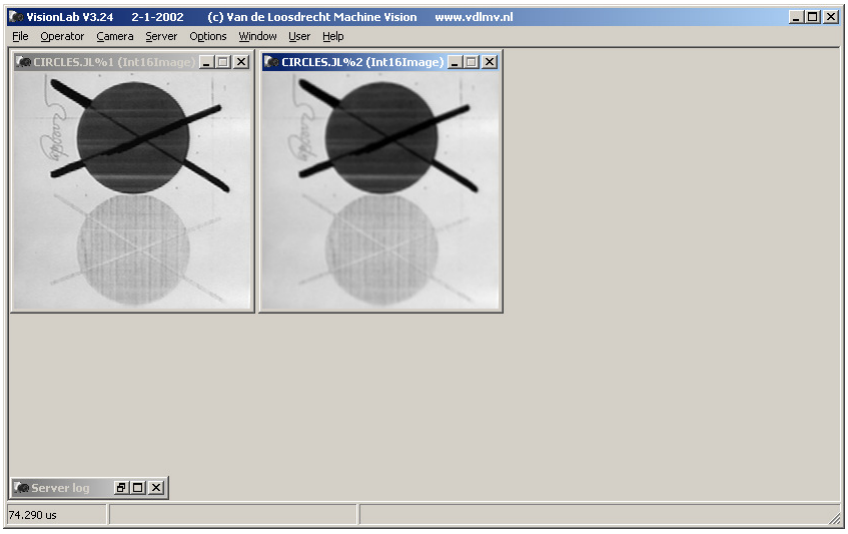
27-aug-18 Linear filters 13

Convolution with smooth mask use EdgeExtend



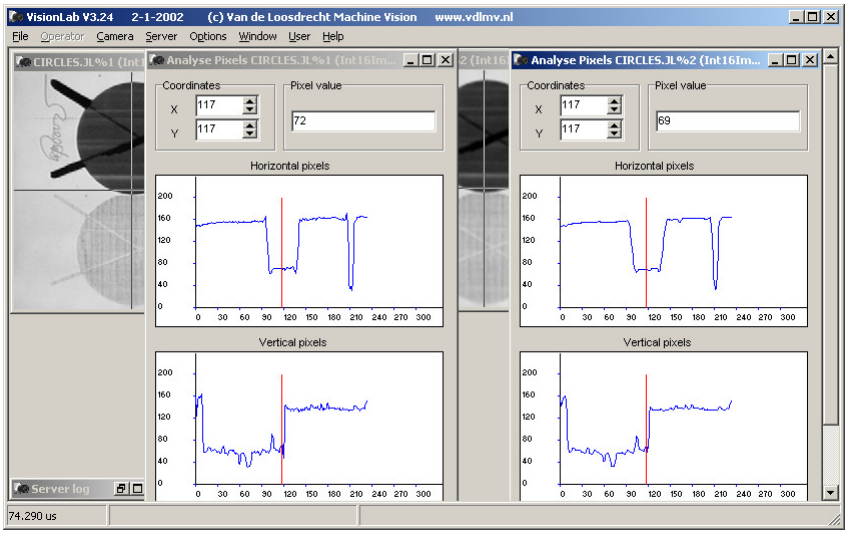
27-aug-18 Linear filters 14

Convolution with smooth mask use EdgeExtend



27-aug-18 Linear filters 15

original **after convolution with smooth**



27-aug-18 Linear filters 16

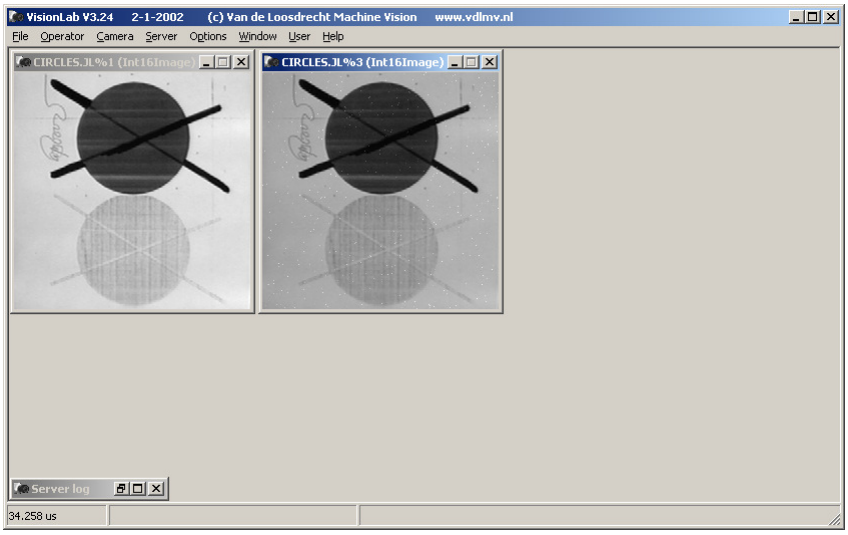
Demonstration noise reduction

Open image circles.jl

- **Add noise 1 0 50 (from Synthetic menu)**
- **Set display LUTs to clip**
- **Remove noise with convolution with gaussian 3x3**
- **Remove noise with convolution with gaussian 5x5, more noise removed but 'softer'**

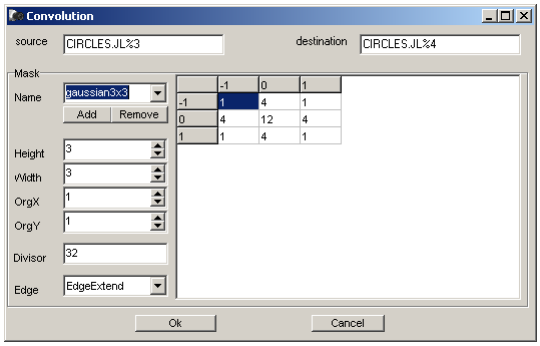
27-aug-18 Linear filters 17

Add noise



27-aug-18 Linear filters 18

Remove noise with convolution with Gaussian 3x3



27-aug-18 Linear filters 19

Remove noise with convolution with Gaussian 3x3

27-aug-18 Linear filters 20

Gaussian Filter

GaussianFilter (image, sigma, size)

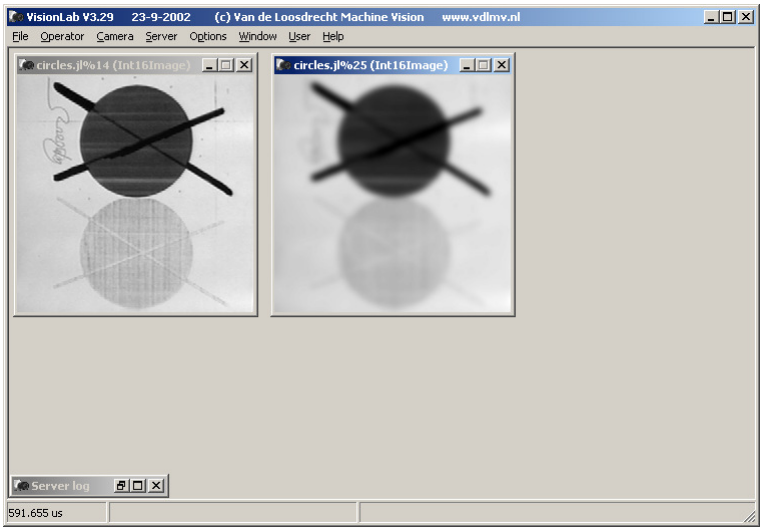
This is a generalised implementation of a Gaussian smoothing filter.

Parameters:

- **sigma is the standard deviation. Typical values are [2/3 .. 3]**
- **size is the size of the neighbourhood of the operation. If size is 0 the algorithm calculates a size so that pixels at 3*sigma are neglected.**

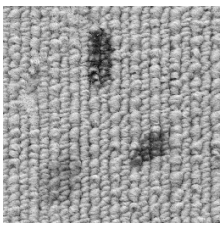
27-aug-18 Linear filters 21

Demonstration Gaussian Filter 3 0



27-aug-18 Linear filters 22

Exercise 1 Gaussian filter

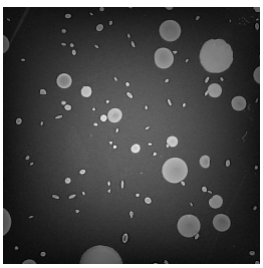


- **Use image carpet.jl**
- **Try to find the dirty spots in the carpet.**

- **answer: carpet.jls**

27-aug-18 Linear filters 23

Exercise 2 Gaussian filter



- **Use image shading_c.jl**
- **Try first to segment the image using Threshold**
- **Then generate a background image using Gaussian and subtract the background before thresholding**

- **answer: gaussian2.jls**

27-aug-18 Linear filters 24

High pass filter

Mask (in general):

- **centre values > 0**
- **ring values < 0**
- **border values = 0**
- **sum values = 0**
- **division factor = 1**

Examples:

- **Laplacian 3x3:**

```
−1   −1   −1
−1    8   −1
−1   −1   −1
```

27-aug-18 Linear filters 25

High pass filter

Examples (continued):

- **Laplacian 5x5:**

```
 0    0   −1    0    0
 0   −1   −2   −1    0
−1   −2   16   −2   −1
 0   −1   −2   −1    0
 0    0   −1    0    0
```

Usage:

- **high pass filter**
- **edge detection, but sensitive to noise**

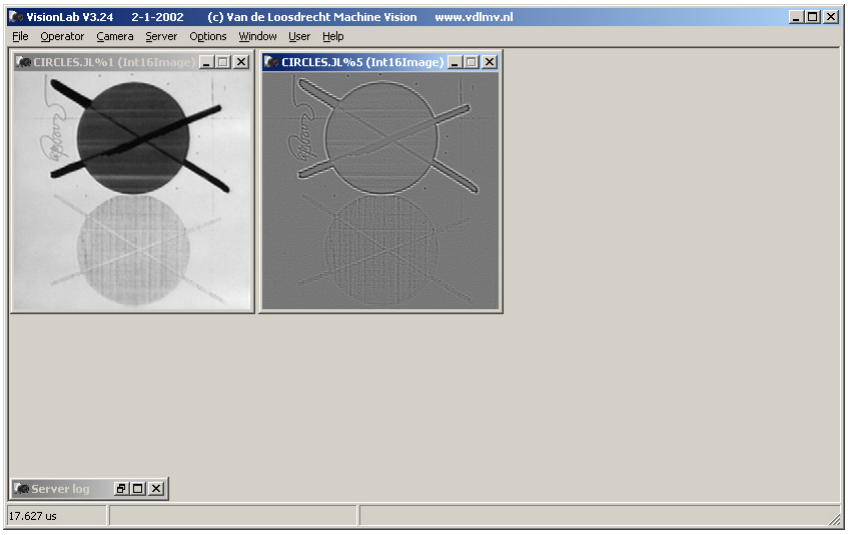
27-aug-18 Linear filters 26

Demonstration high pass filter

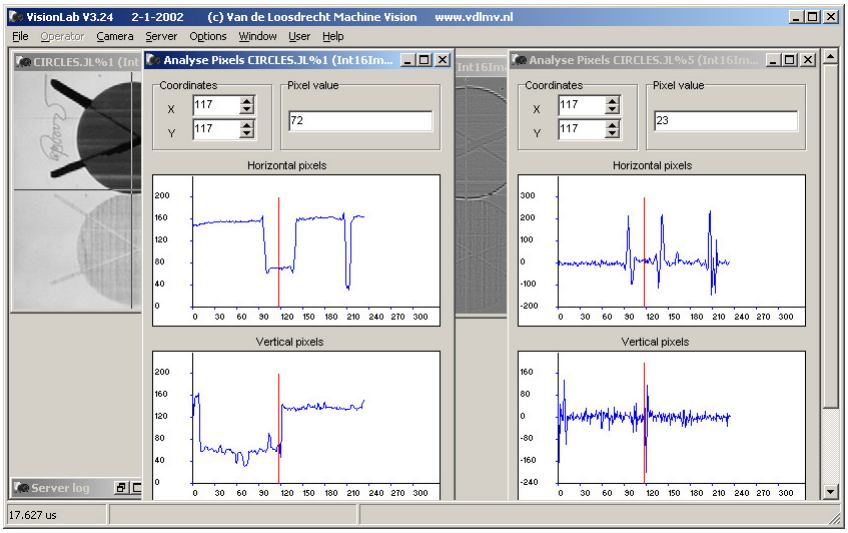
- Open image circles.jl
- Convolution with Laplacian 3x3
- 2x analyse pixels:
  - low frequencies -> 0
  - high frequencies -> |pixel value| >> 1
- Convolution with Laplacian 3x3 on image with noise 1 0 50
- Note: noise is amplified
- Can be used as edge detection: Threshold 40 10000 see next example

- Convolution Laplacian 5x5 on circles
- Threshold 40 10000 gives sharp edges

Convolution with Laplacian 3x3

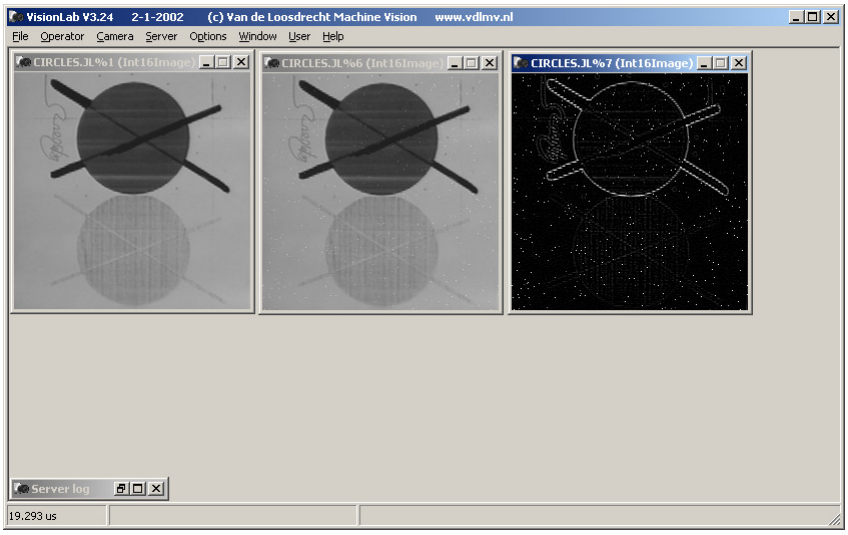


original **after convolution with Laplacian**



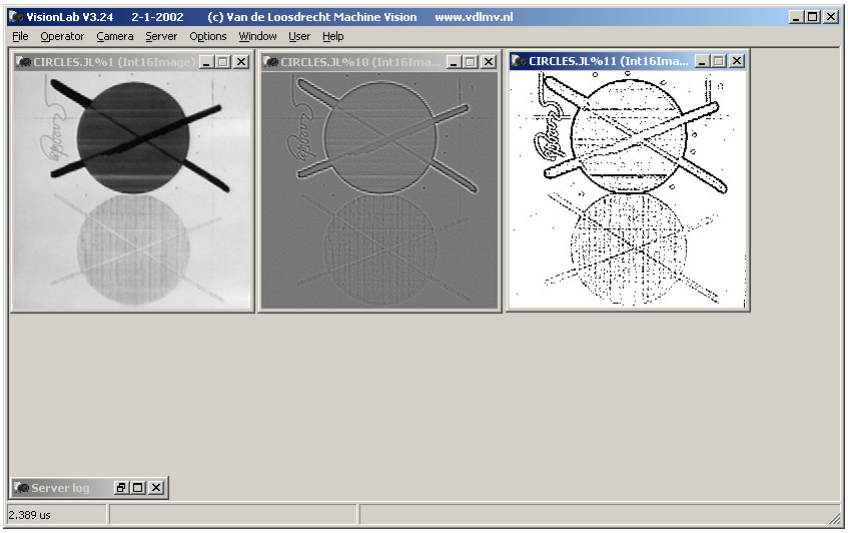
27-aug-18 Linear filters 29

Convolution with Laplacian after adding noise



27-aug-18 Linear filters 30

Edge detection: convolution with Laplacian 5x5 succeeded by threshold 40 10000



27-aug-18 Linear filters 31

Sharpening (*)

- **Variant of high pass filter sharpening:**
  Laplacian but original image is 'added' to result
- **Mask:**

```
1   1   1
1  -7   1
1   1   1
```

  (inverted Laplacian + '1')
- **Digital variant of dark room technique in use by TV and video recorder**

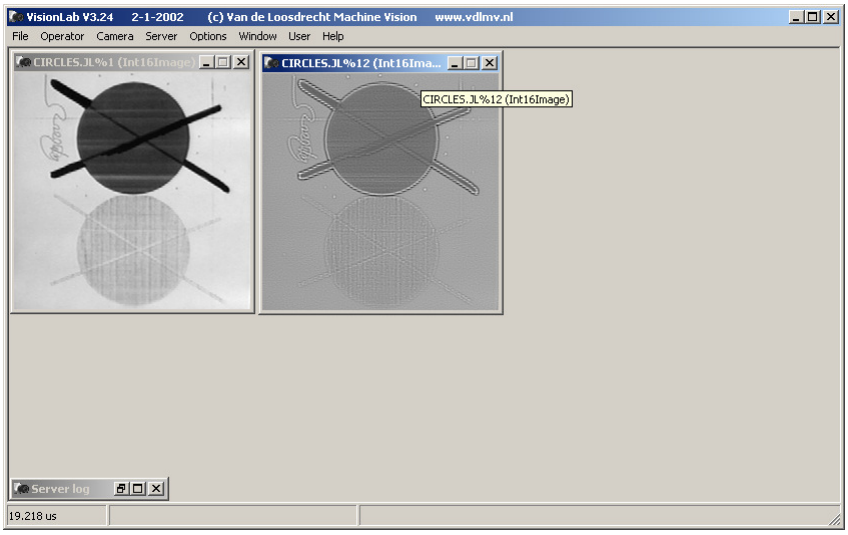
27-aug-18 Linear filters 32

Demonstration sharpening (*)

- Open image circles.jl
- Convolution with sharpening mask
- Convolution with Laplacian 3x3
- Add original to laplacianed image
- Explain small difference due to division factor
- Dark room technique:
  - Divide result convolution with Laplacian3x3 by pixel value 5
  - Add it to original image

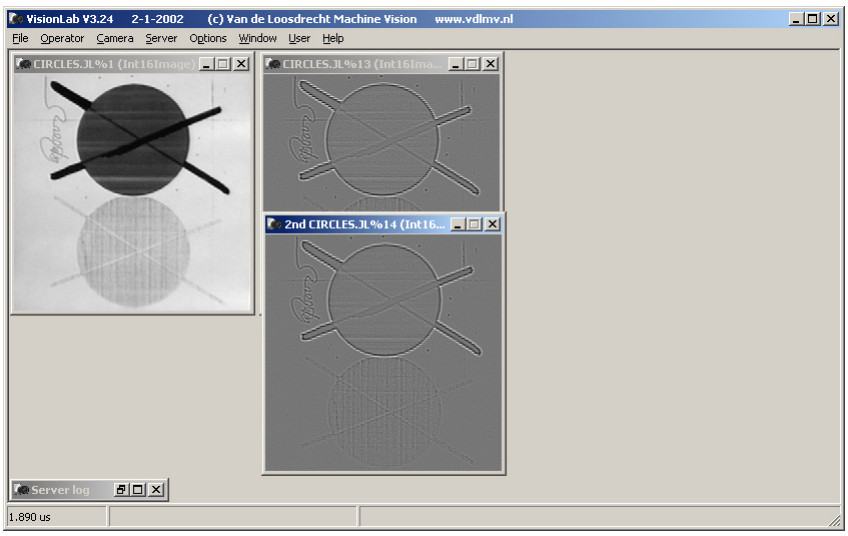
27-aug-18 Linear filters 33

Convolution with sharpening mask (*)



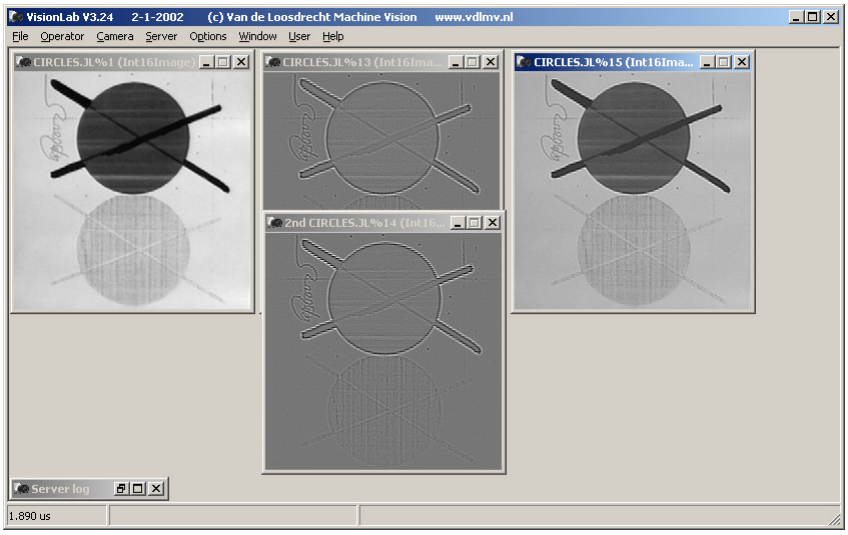
27-aug-18 Linear filters 34

Divide result convolution with Laplacian3x3 by pixel value 5 (*)



27-aug-18 Linear filters 35

Add it to original image (*)



27-aug-18 Linear filters 36

Exercise sharpening (*)

Make a script which sharpens the image
(optional: add this script as an operator to the system)

The script should accept three parameters (image, result_image and divide factor) perform the operations and display the result

Answer: sharpen.jls and sharpen.ini

Mexican hat (*)

- **Combination of low and high pass filter**
- **A Laplacian smoothed by a Gaussian**
- **Mask (7x7):**

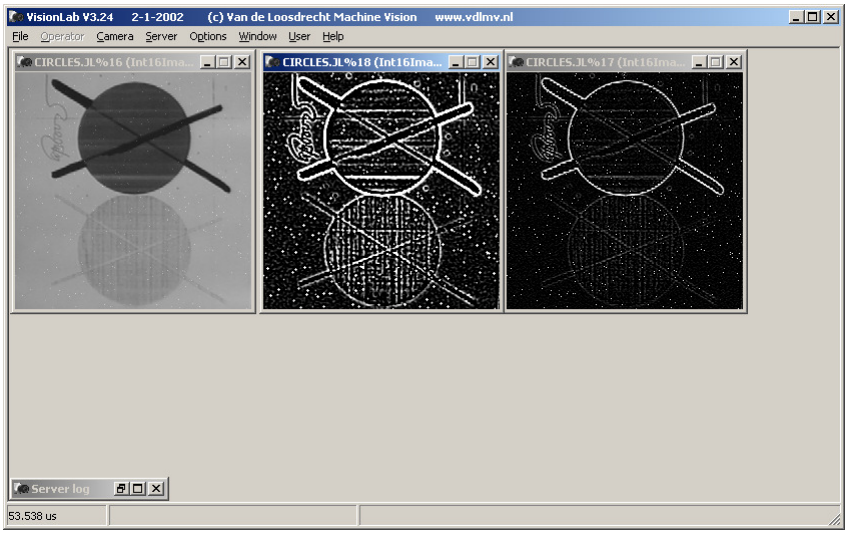
```
 0   0  -1  -1  -1   0   0
 0  -1  -3  -3  -3  -1   0
-1  -3   0   7   0  -3  -1
-1  -3   7  24   7  -3  -1
-1  -3   0   7   0  -3  -1
 0  -1  -3  -3  -3  -1   0
 0   0  -1  -1  -1   0   0
```

- **Local noise is smoothed out by low pass filter in centre**

Demonstration Mexican hat (*)

- **Open image circles.jl**
- **Add noise 1 0 50**
- **Convolution Laplacian 5x5 on noise image**
- **Convolution Mexican hat on noise image (smooth noise and enhance high frequencies)**

With noise **Mexican hat** **Laplacian (*)**



Laplacian of Gaussian (LoG) filter (*)

LoGFilter (image, sigma, size)

This is a generalised implementation of a Mexican hat filter

Parameters:

- **sigma is the standard deviation. Typical values are [2/3 .. 10]**
- **size is the size of the neighbourhood of the operation. If size is 0 the algorithm calculates a size so that pixels at 3*sigma are neglected**

27-aug-18 Linear filters 41

Difference of Gaussians (DoG) filter (*)

DoGFilter (image, sigmaLow, sigmaHigh, size)

An alternative implementation for a generalised Mexican hat filter, using the difference of two Gaussians with substantially different sigmas.

Parameters:

- **sigmaLow and sigmaHigh are the standard deviations for the DoG operator. Typical values are [0 .. 10].**
  For good approximation Mexican hat of the ratio sigmaHigh:sigmaLow should be 1.6:1.
- **size is the size of the neighbourhood of the operation. If size is 0 the algorithm calculates a size so that pixels at 3*sigma are neglected**

27-aug-18 Linear filters 42

Edge detection

Mask (in general):

- **centre value = 0**
- **sum of values = 0**
- **division factor = 1**

Examples:

- **Sobel NS:**

```
−1   −2   −1
 0    0    0
 1    2    1
```


Edge detection

Examples (continued):

- **Sobel WE:**

```
−1    0    1
−2    0    2
−1    0    1
```

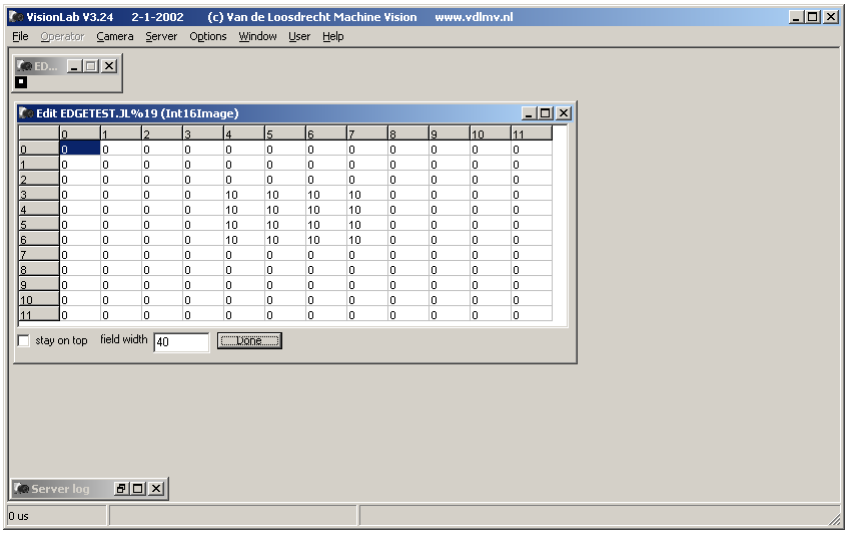
Introduction to the next chapter about edge detection

Demonstration edge detection

- Open image edgetest.jl *(do not use image sq2.jl)*
- Show with edit pixel values of image
- Convolution with SobelNS mask
- Show with edit pixel values of result
- Show mask without executing convolution
- Explanation:
  transitions from 0 (up) to 10 (down) gives positive values
  transitions from 10 (up) to 0 (down) gives negative values
  transitions from left to right and vs. give zeros

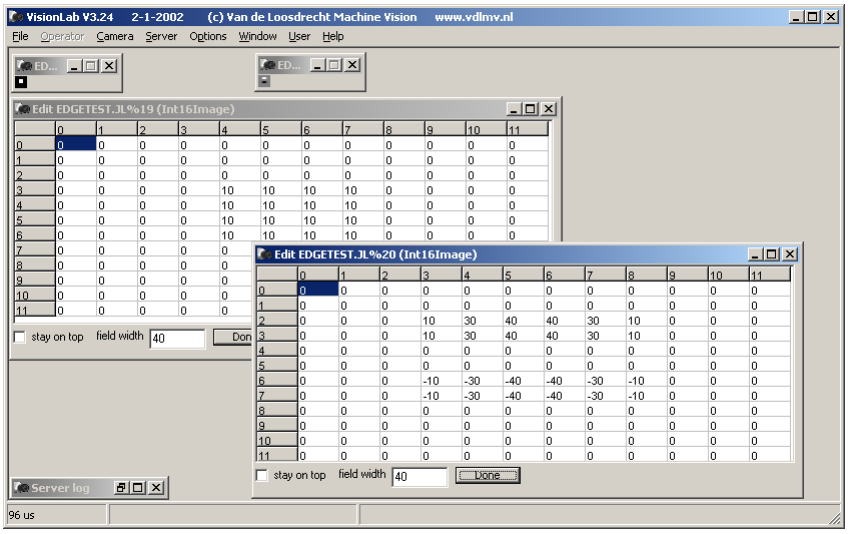
27-aug-18 Linear filters 45

Image edgetest.jl



27-aug-18 Linear filters 46

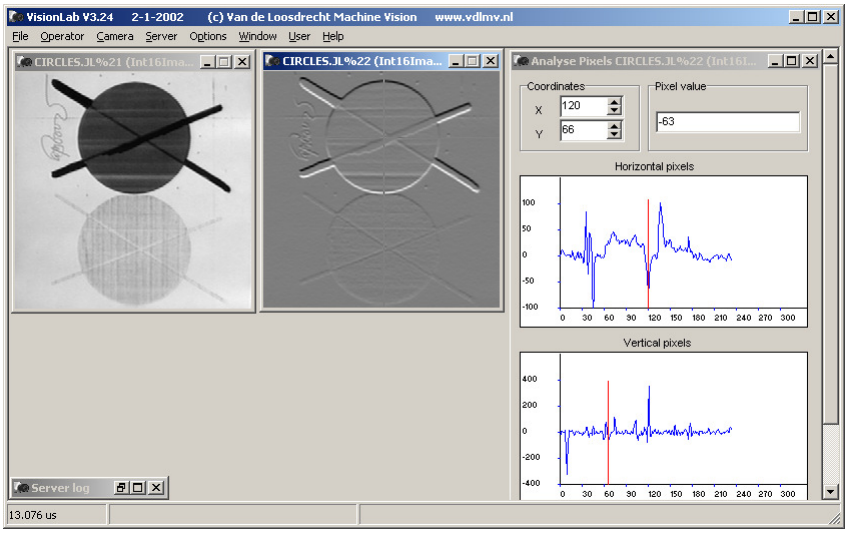
Convolution with SobelNS mask



Demonstration edge detection

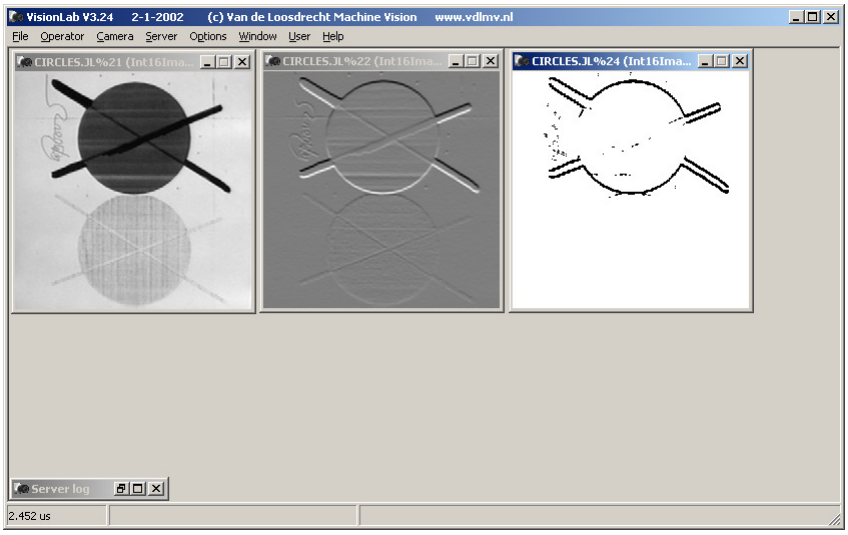
- **Open image circles.jl**
- **Convolution with SobelNS mask**
- **Analyse pixels:**
  - **horizontal edges are strong**
  - **vertical edges are weak**
- **Threshold 100 -100 to show the edges**
- **Convolution with SobelWE mask on circles for vertical edges**

Convolution with SobelNS mask



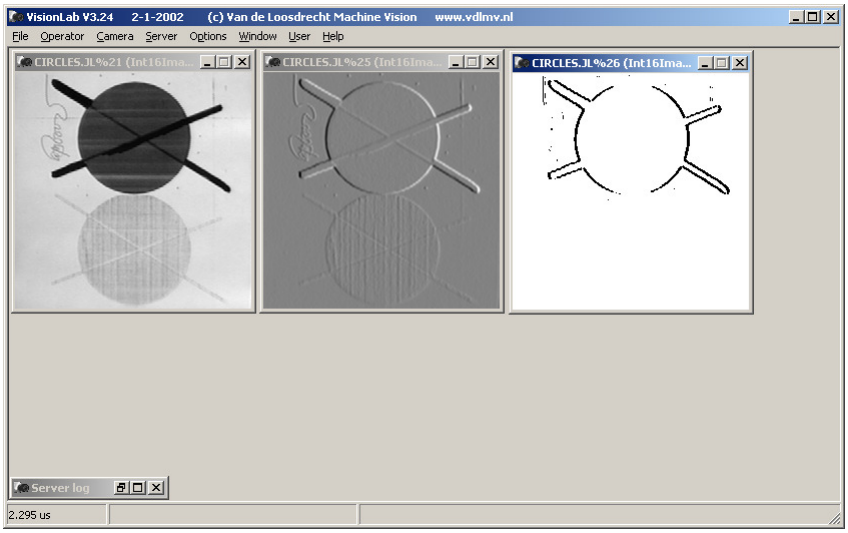
27-aug-18 Linear filters 49

Threshold 100 -100 to show the edges



27-aug-18 Linear filters 50

Convolution with SobelWE mask followed by threshold



27-aug-18 Linear filters 51