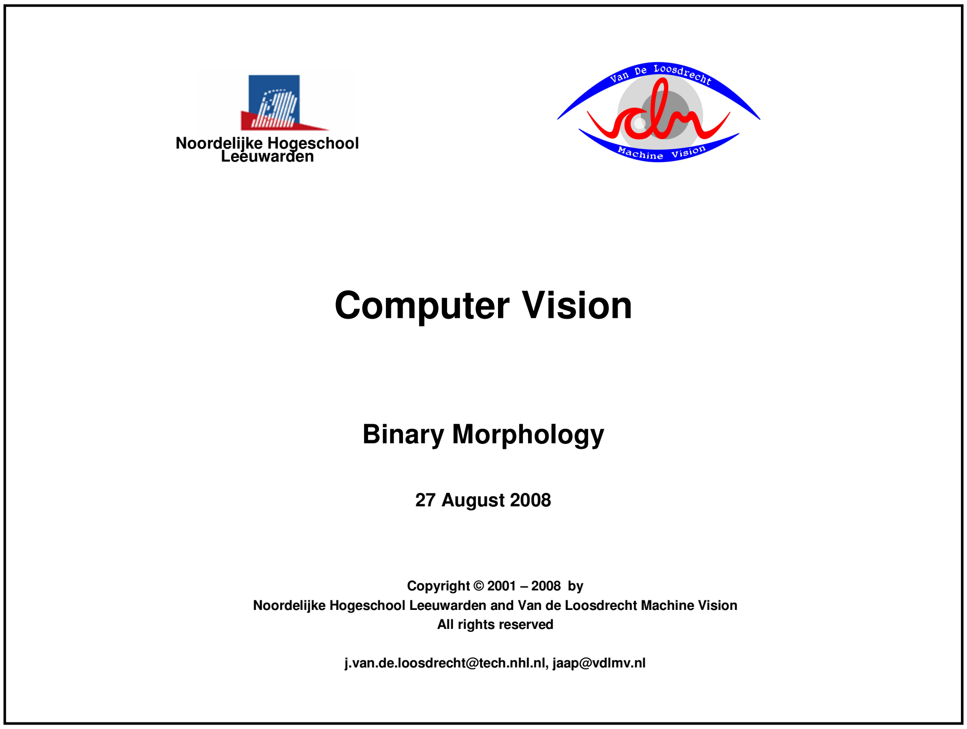


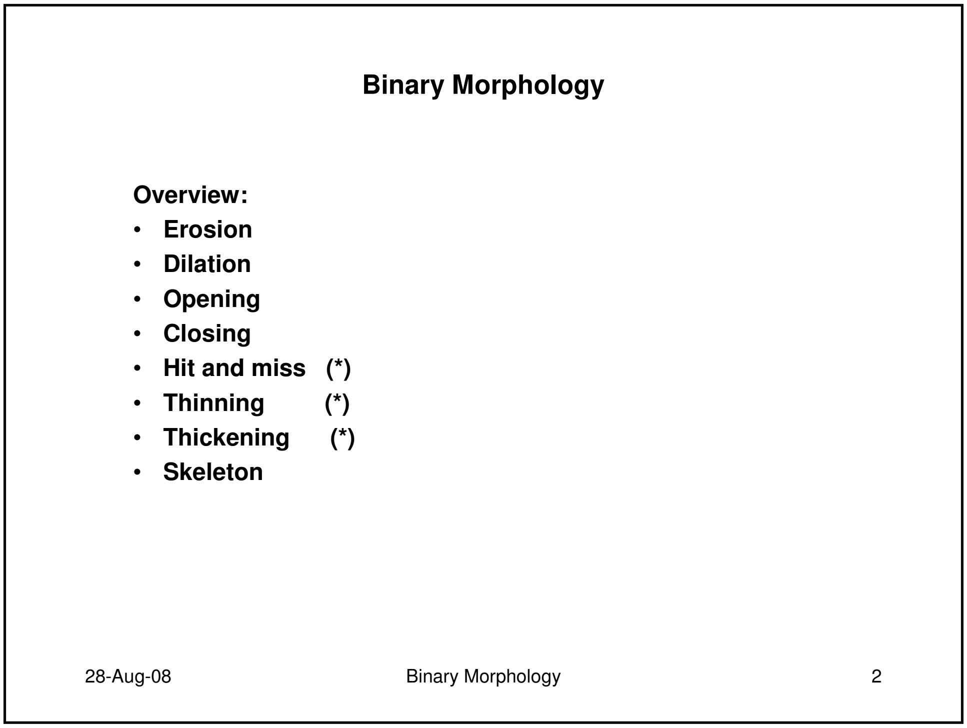
Noordelijke Hogeschool
Leeuwarden

Computer Vision

Binary Morphology

27 August 2008

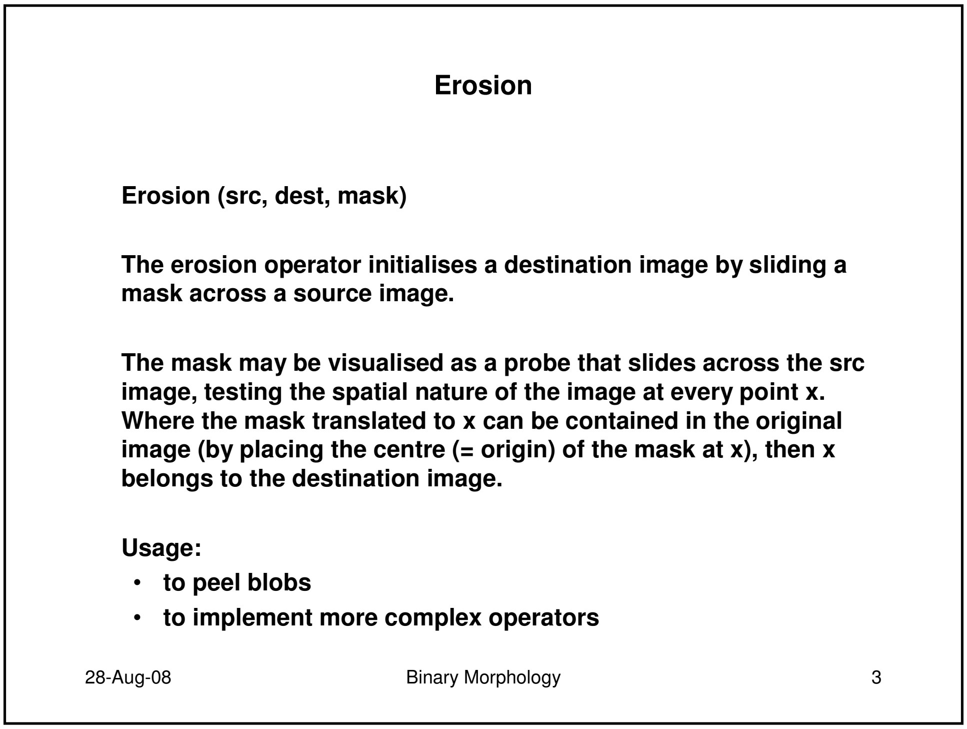
j.van.de.loosdrecht@tech.nhl.nl, jaap@vdlmv.nl



Binary Morphology

Overview:

- **Erosion**
- **Dilation**
- **Opening**
- **Closing**
- **Hit and miss (*)**
- **Thinning (*)**
- **Thickening (*)**
- **Skeleton**



Erosion

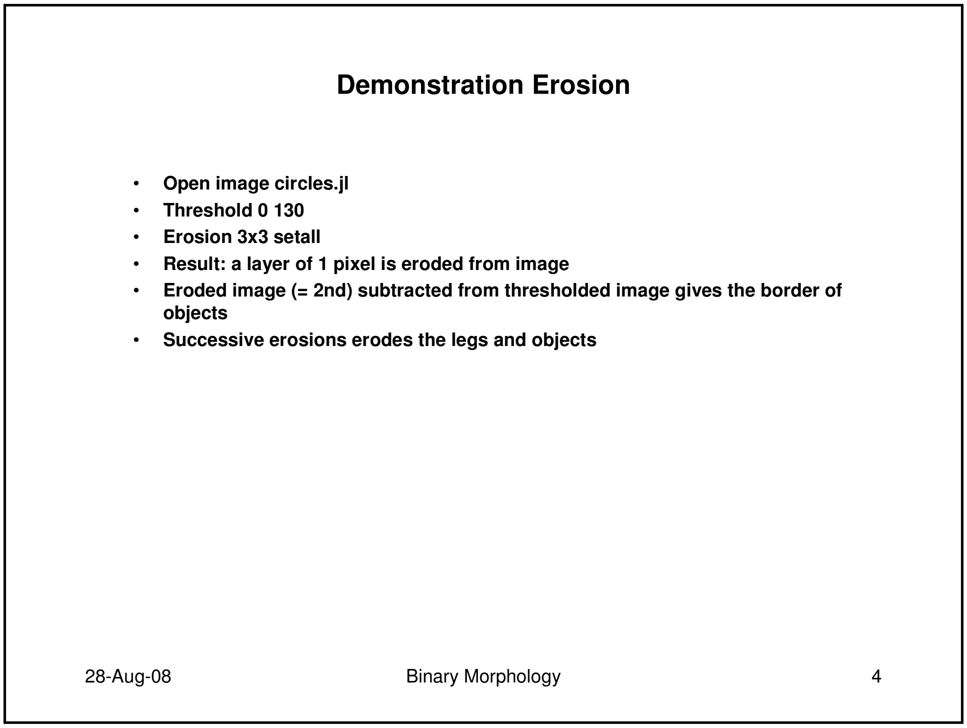
Erosion (src, dest, mask)

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

The mask may be visualised as a probe that slides across the src image, testing the spatial nature of the image at every point x. Where the mask translated to x can be contained in the original image (by placing the centre (= origin) of the mask at x), then x belongs to the destination image.

Usage:

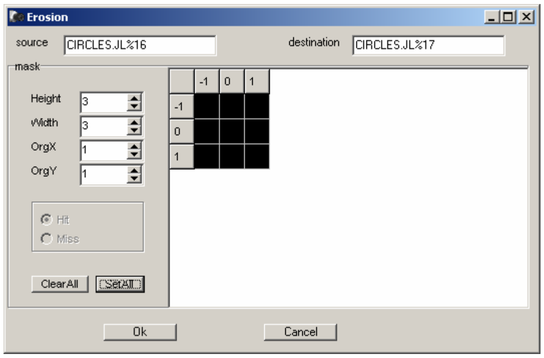
- **to peel blobs**
- **to implement more complex operators**



Demonstration Erosion

- **Open image circles.jl**
- **Threshold 0 130**
- **Erosion 3x3 setall**
- **Result: a layer of 1 pixel is eroded from image**
- **Eroded image (= 2nd) subtracted from thresholded image gives the border of objects**
- **Successive erosions erodes the legs and objects**

Erosion 3x3 setall



Erosion

source CIRCLES.JL%16

destination CIRCLES.JL%17

mask

Height 3

Width 3

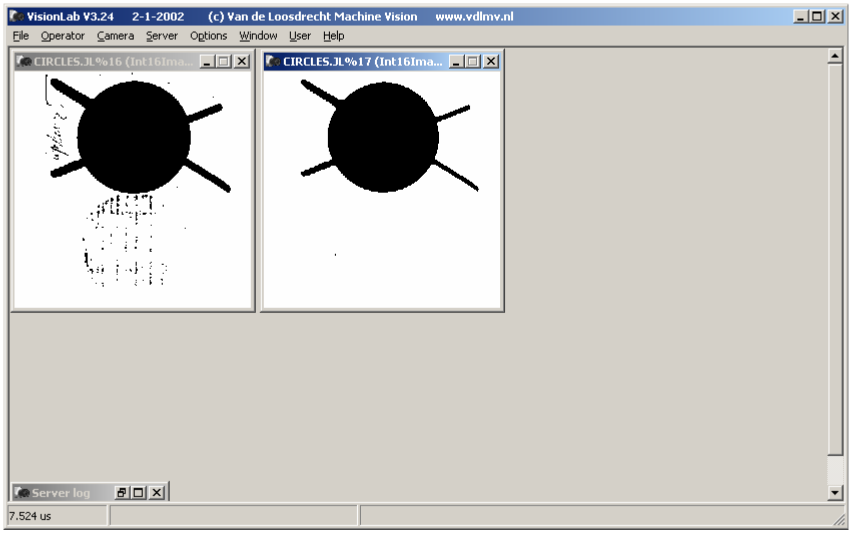
OrgX 1

OrgY 1

● Hit
○ Miss

	-1	0	1
-1	■	■	■
0	■	■	■
1	■	■	■

Erosion 3x3 setall



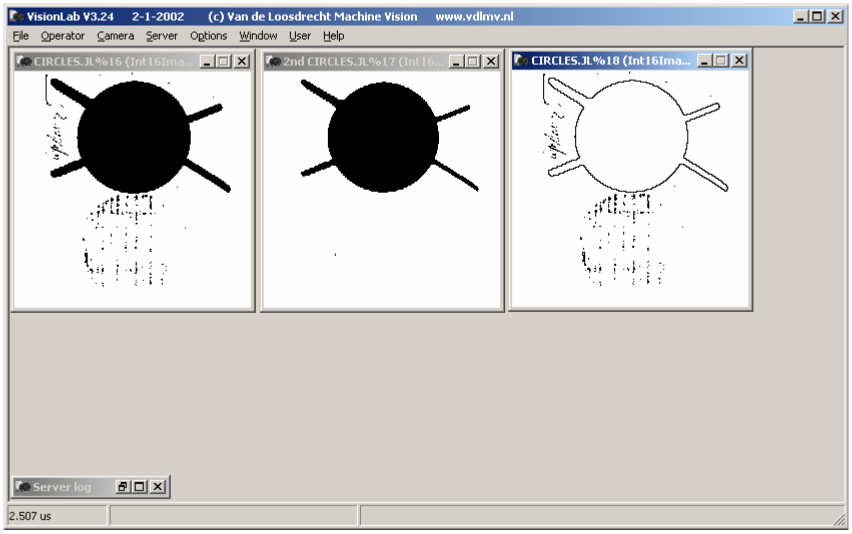
VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl

CIRCLES.JL%16 (Int16Ima...

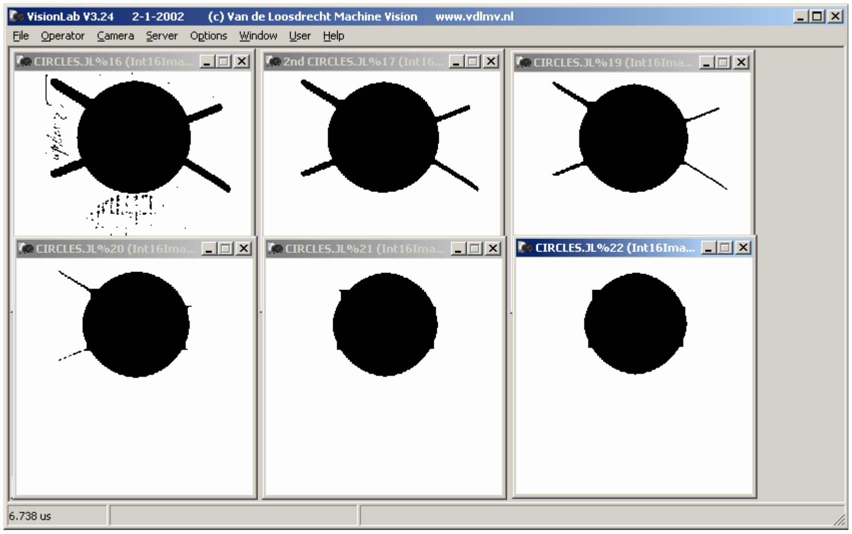
CIRCLES.JL%17 (Int16Ima...

7.524 us

Eroded image (= 2nd) subtracted from Thresholded image gives the border of objects


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
CIRCLES.JL%16 (Int16Ima...
2nd CIRCLES.JL%17 (Int16...
CIRCLES.JL%18 (Int16Ima...
Server log
2.507 us

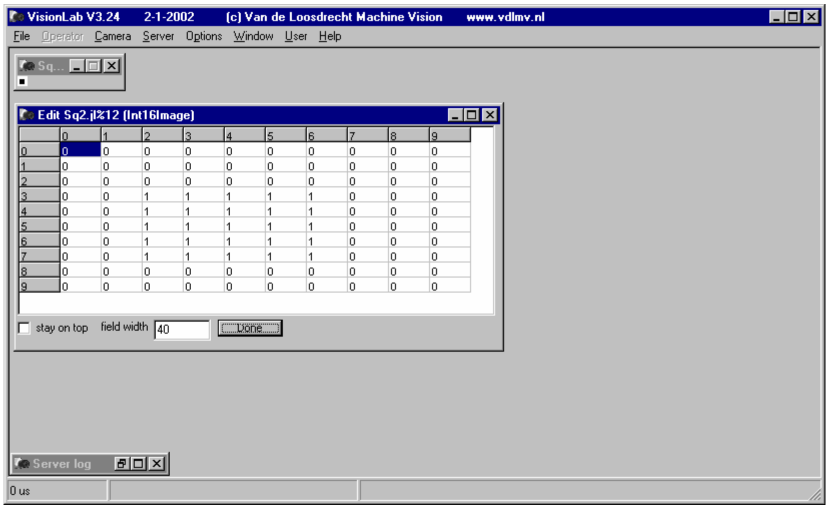
Successive erosions erodes the legs and objects


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
CIRCLES.JL%16 (Int16Ima...
2nd CIRCLES.JL%17 (Int16...
CIRCLES.JL%19 (Int16Ima...
CIRCLES.JL%20 (Int16Ima...
CIRCLES.JL%21 (Int16Ima...
CIRCLES.JL%22 (Int16Ima...
6.738 us

Demonstration Erosion

- Open image sq2.jl
- Show image with edit
- Erosion with 1 1 1
  0 0 1
  0 0 0
- Show result with edit
- Explain why top, right and left border are removed

Image sq2.jl

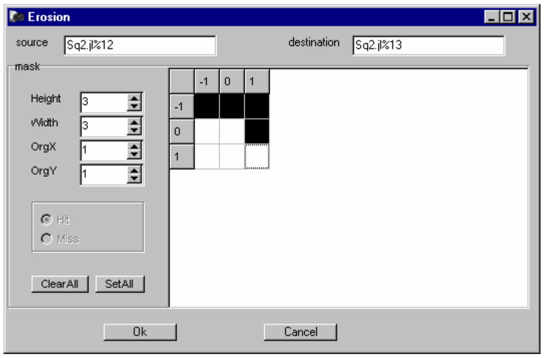

VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Sq...
Server log
0 us

Edit Sq2.jl%12 (Int16Image)

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	1	1	1	1	1	0	0	0
4	0	0	1	1	1	1	1	0	0	0
5	0	0	1	1	1	1	1	0	0	0
6	0	0	1	1	1	1	1	0	0	0
7	0	0	1	1	1	1	1	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

□ stay on top field width 40 Done

Erosion mask



source Sq2.jl%12

destination Sq2.jl%13

mask

Height 3

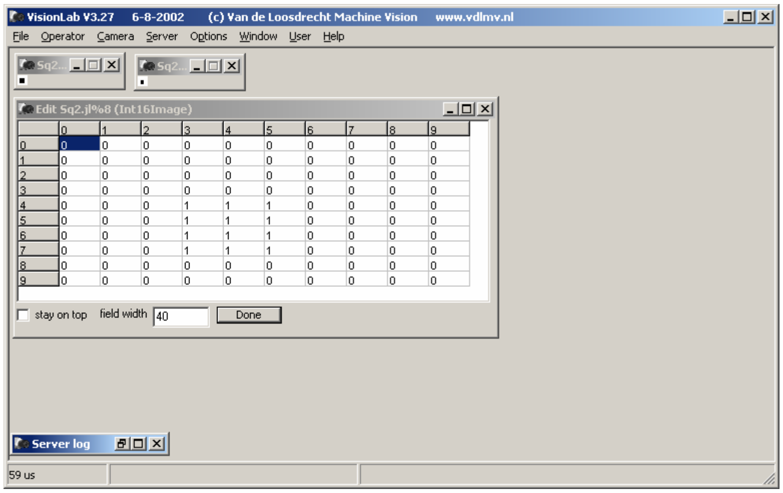
Width 3

OrgX 1

OrgY 1

	-1	0	1
-1	■	■	■
0	□	□	■
1	□	□	□

Erosion


VisionLab V3.27 6-8-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File
Operator
Camera
Server
Options
Window
User
Help
Sq2...
Sq2...
Server log
59 us

Edit Sq2.jl%8 (Int16Image)

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	1	1	1	0	0	0	0
5	0	0	0	1	1	1	0	0	0	0
6	0	0	0	1	1	1	0	0	0	0
7	0	0	0	1	1	1	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

□ stay on top field width 40 Done

Dilation

Dilation (src, dest, mask)

At the start of the operation the destination image is filled with background (=0) pixel values. The mask is swept over the source image. Every time the centre (= origin) of the mask touches an object (=1) pixel in the source image, the entire translated mask is OR-ed to the destination image.

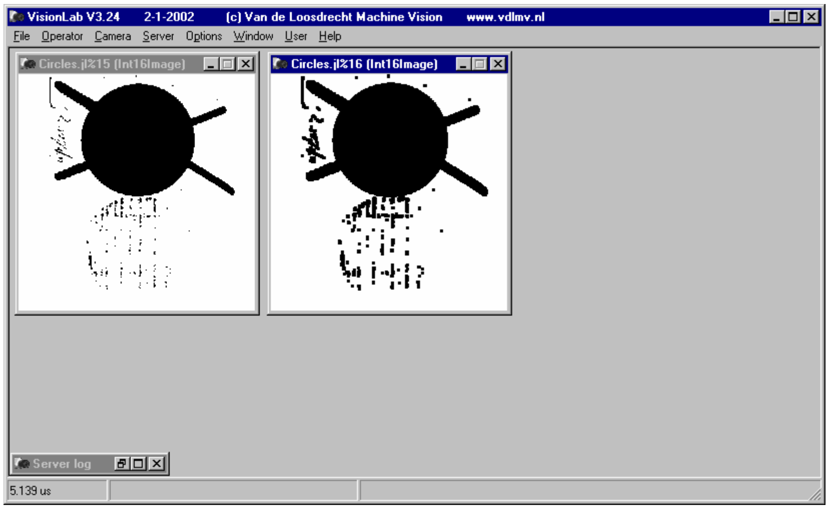
Usage:

- **to add a layer to blobs**
- **to implement more complex operators**

Demonstration Dilation

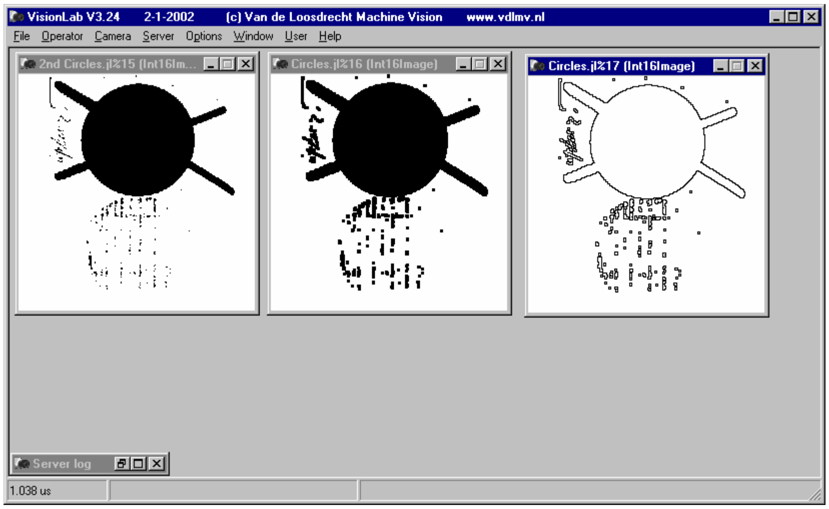
- **Open image circles.jl**
- **Threshold 0 130**
- **Dilation 3x3 setall**
- **Result: a layer of 1 pixel is added to the image**
- **Thresholded image (= 2nd) subtracted from Eroded image gives a border around the objects**
- **With successive dilations the objects grow**

Dilation 3x3 setall


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Circles.jl%15 (Int16Image)
Circles.jl%16 (Int16Image)
Server log
5.139 us

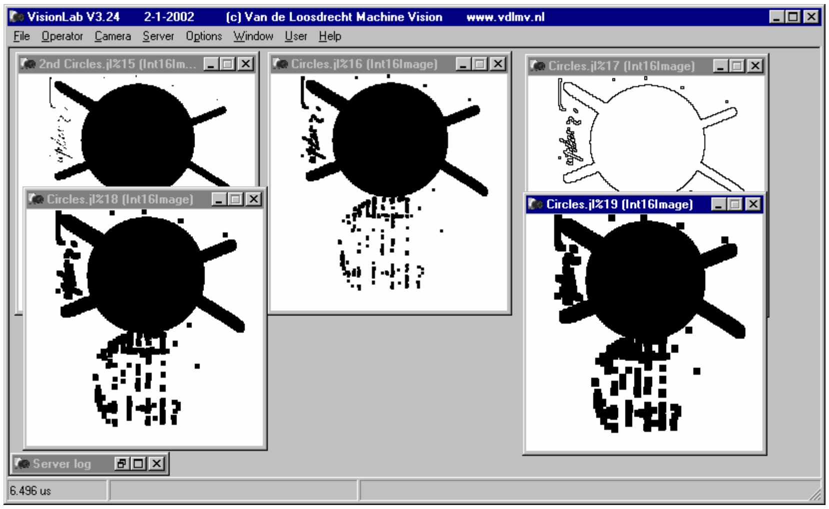
28-Aug-08 Binary Morphology 15

Thresholded image (= 2nd) subtracted from Eroded image gives a border around the objects


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
2nd Circles.jl%15 (Int16Im...
Circles.jl%16 (Int16Image)
Circles.jl%17 (Int16Image)
Server log
1.038 us

28-Aug-08 Binary Morphology 16

With successive dilations the objects grow

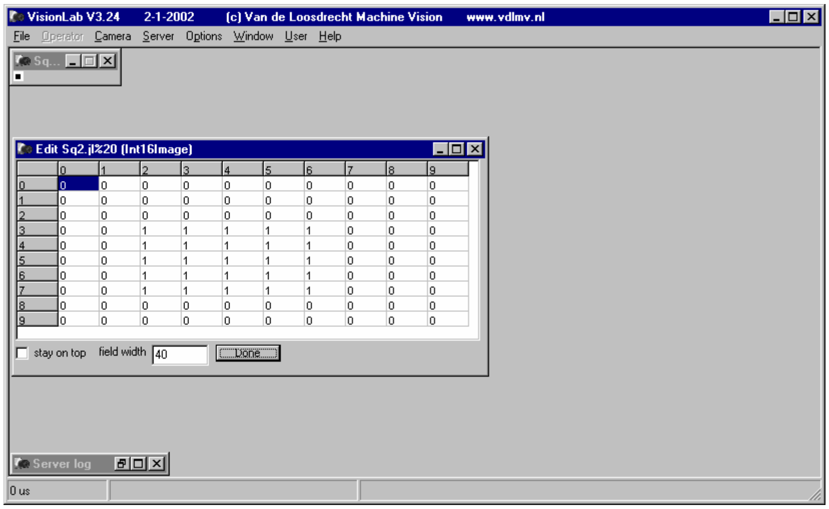

VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
2nd Circles.jl%15 (Int16Im...
Circles.jl%16 (Int16Image)
Circles.jl%17 (Int16Image)
Circles.jl%18 (Int16Image)
Circles.jl%19 (Int16Image)
Server log
6.496 us

Demonstration Dilation

- Open image sq2.jl
- Show image with edit
- Dilation with 0 0 0
  0 0 0
  1 0 0
- Show result with edit
- Explain why square is shifted one square to left bottom

- Same with Dilation with 0 0 0, result extra layer at left side
  1 1 0
  0 0 0

Image sq2.jl

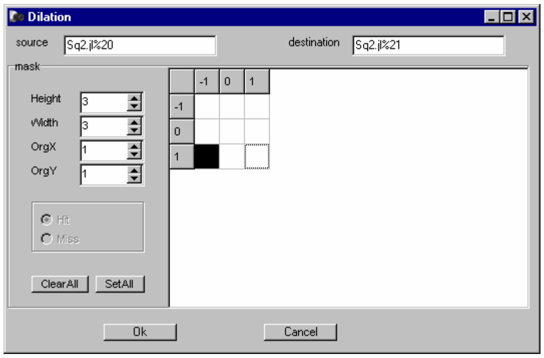

VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Sq...
Server log
0 us

Edit Sq2.jl%20 (Int16Image)

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	1	1	1	1	1	0	0	0
4	0	0	1	1	1	1	1	0	0	0
5	0	0	1	1	1	1	1	0	0	0
6	0	0	1	1	1	1	1	0	0	0
7	0	0	1	1	1	1	1	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

□ stay on top field width 40 Done

First example Dilation



source Sq2.jl%20

destination Sq2.jl%21

mask

Height 3

Width 3

OrgX 1

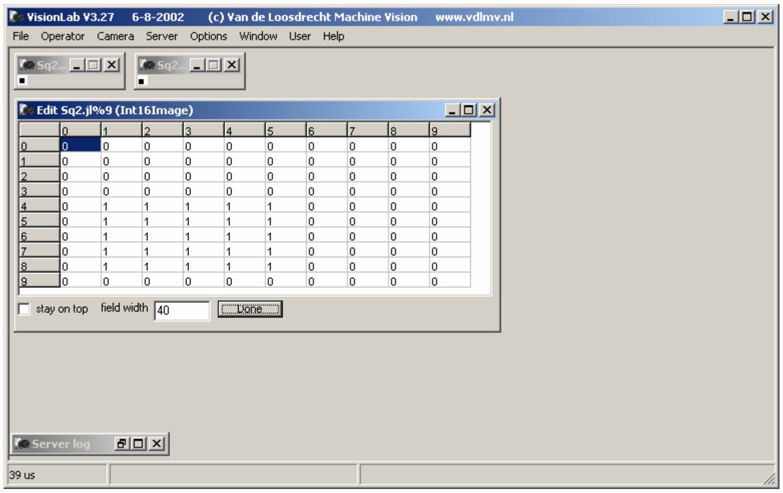
OrgY 1

● Hit

○ Miss

	-1	0	1
-1			
0			
1	■		

First example Dilation



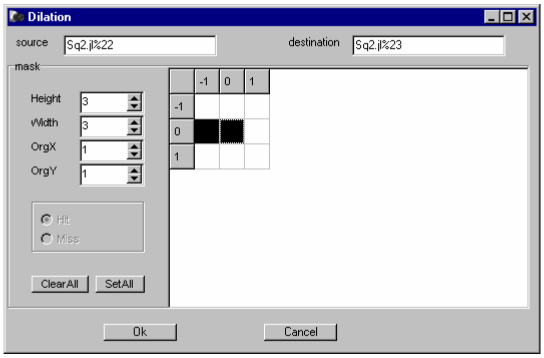
VisionLab V3.27 6-8-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Sq2...
Sq2...
Server log
39 us

Edit Sq2.jl%9 (Int16Image)

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	1	1	1	1	1	0	0	0	0
5	0	1	1	1	1	1	0	0	0	0
6	0	1	1	1	1	1	0	0	0	0
7	0	1	1	1	1	1	0	0	0	0
8	0	1	1	1	1	1	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

□ stay on top field width 40 Done

Second example Dilation



source Sq2.jl%22 destination Sq2.jl%23

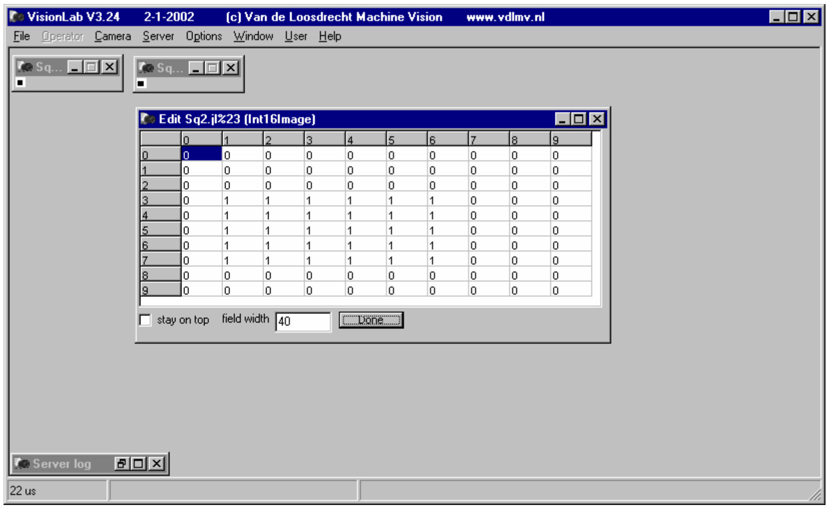
mask

Height 3
Width 3
OrgX 1
OrgY 1

● Hit
○ Miss

	-1	0	1
-1			
0	■	■	
1			

Second example Dilation


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Sq...
Sq...
Edit Sq2.jl%23 (Int16Image)
Done
Server log
22 us

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	1	1	1	1	1	1	0	0	0
4	0	1	1	1	1	1	1	0	0	0
5	0	1	1	1	1	1	1	0	0	0
6	0	1	1	1	1	1	1	0	0	0
7	0	1	1	1	1	1	1	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

☐ stay on top field width 40

Opening

Opening (src, dest, mask)

The opening operator initialises a destination image as the result of an erosion on the source image with the mask followed by a dilation with the same mask.

Usage: smooths the contour by

- **breaking narrow corridors**
- **eliminating small islands**
- **sharpening peaks**
- **small objects are removed and the original shape is almost retained**

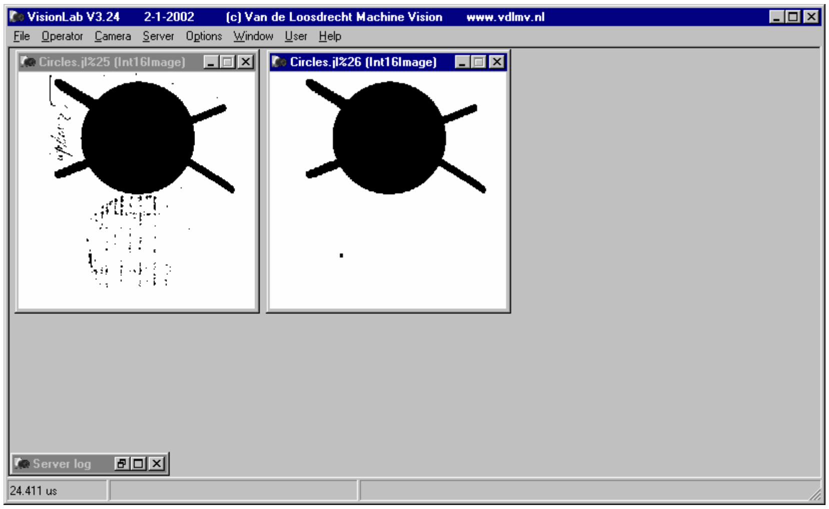
Opening operation is idempotent

Demonstration Opening, removing small objects

- **Open image circles.jl**
- **Threshold 0 130**
- **Opening 3x3 setall**
- **Note small difference: difference between thresholded image and Opened image, right top leg, right edge (207,35), due to square shape erosion mask (*)**
- **2nd time same opening has no effect**

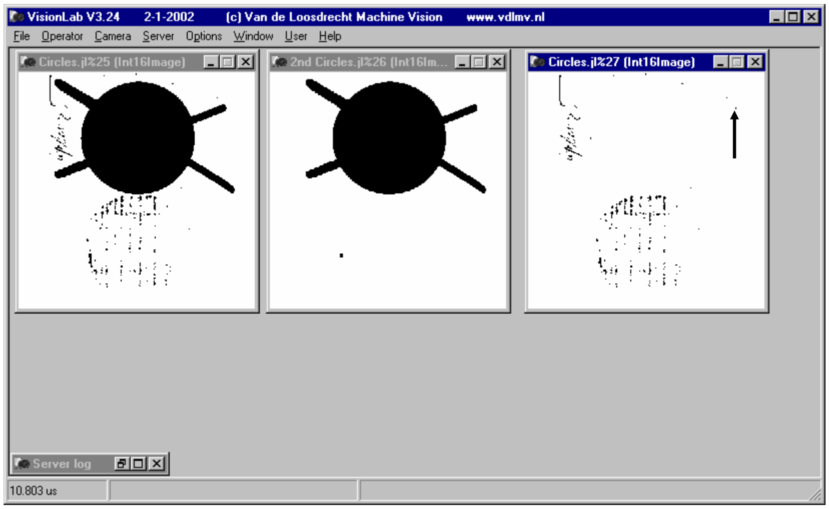
28-Aug-08 Binary Morphology 25

Opening 3x3 setall, removing small objects


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Circles.jl%25 (Int16Image)
Circles.jl%26 (Int16Image)
Server log
24.411 us

28-Aug-08 Binary Morphology 26

Difference


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Circles.jl%25 (Int16Image)
2nd Circles.jl%26 (Int16Im...
Circles.jl%27 (Int16Image)
Server log
10.803 us

28-Aug-08 Binary Morphology 27

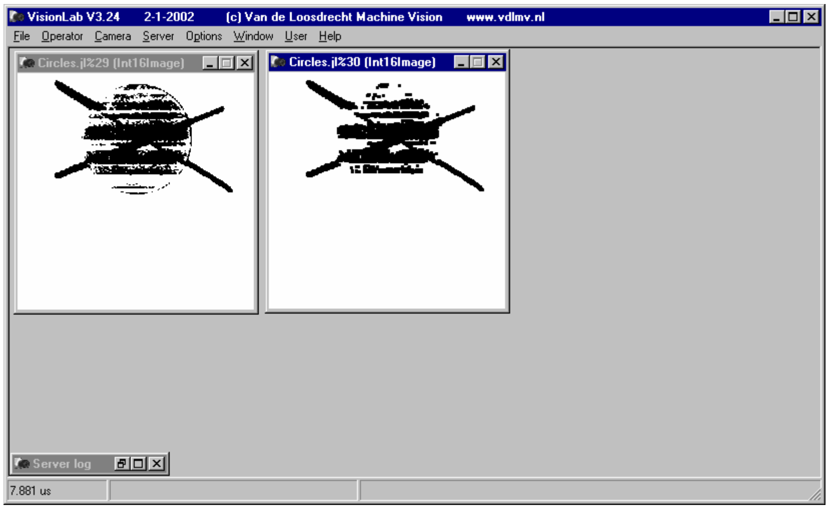
Demonstration Opening, sharpening of peaks

- **Open image circles.jl**
- **Threshold 0 60**
- **Opening 3x3 setall**
- **Show sharpening of peaks**

- **Implementation:**
  - **Erosion, followed by**
  - **Dilation with same mask**

28-Aug-08 Binary Morphology 28

Opening 3x3 setall, sharpening of peaks


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Circles.jl%29 (Int16Image)
Circles.jl%30 (Int16Image)
Server log
7.881 us

28-Aug-08 Binary Morphology 29

Closing

Closing (src, dest, mask)

The closing operator initialises a destination image as the result of a dilation on the source image with the mask followed by an erosion with the same mask.

Usage: smooths the contour by

- **filling narrow breaks**
- **eliminating small holes**
- **filling gaps**

Closing operation is idempotent

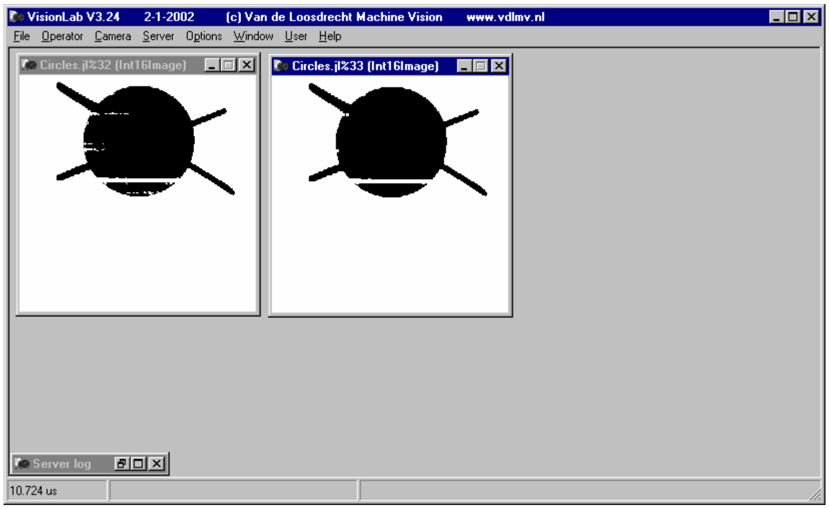
28-Aug-08 Binary Morphology 30

Demonstration Closing, filling gaps

- **Open image circles.jl**
- **Threshold 0 70**
- **Closing 3x3 setall**
- **2nd time same closing has no effect**

- **Implementation:**
  - **Dilation, followed by**
  - **Erosion with same mask**

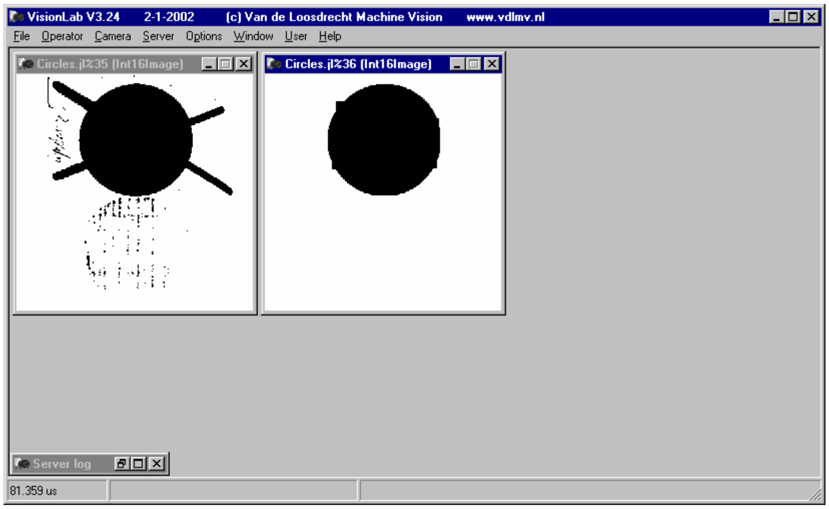
Closing 3x3 setall, filling gaps


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Circles.jl%32 (Int16Image)
Circles.jl%33 (Int16Image)
Server log
10.724 us

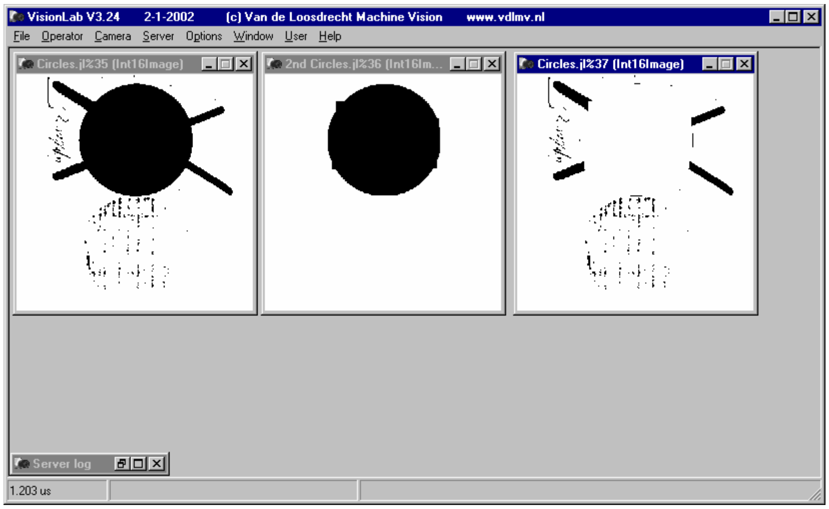
Demonstration Opening, find legs

- Find the legs in threshold circles.jl 0 130:
- Opening 15x15 setall (org = 7,7), remove noise and legs
- Subtract Opened image (= 2nd) from thresholded image
- Opening 5x5 setall (org = 2,2) to remove noise

Opening 15x15 setall (org = 7,7), remove noise and legs

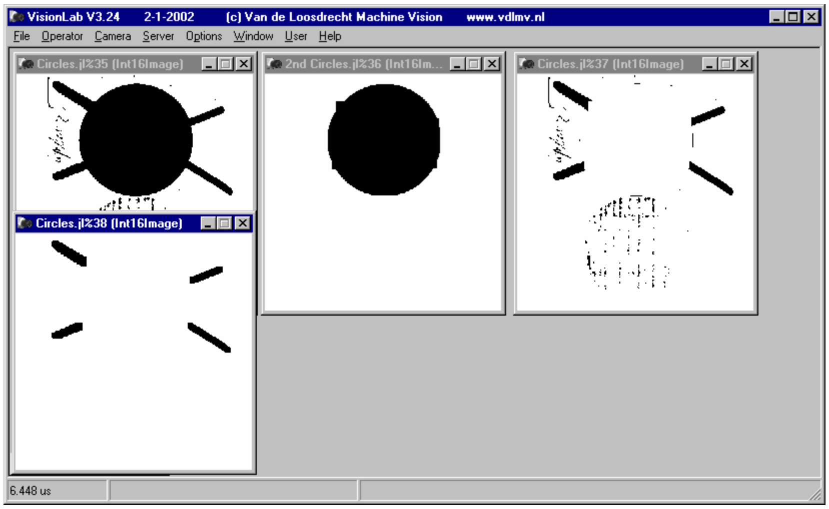

VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Circles.jl%35 (Int16Image)
Circles.jl%36 (Int16Image)
Server log
81.359 us

Subtract Opened image (= 2nd) from Thresholded image


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Circles.jl%35 (Int16Image)
2nd Circles.jl%36 (Int16Im...
Circles.jl%37 (Int16Image)
Server log
1.203 us

28-Aug-08 Binary Morphology 35

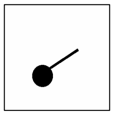
Opening 5x5 setall (org = 2,2) to remove noise


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Circles.jl%35 (Int16Image)
2nd Circles.jl%36 (Int16Im...
Circles.jl%37 (Int16Image)
Circles.jl%38 (Int16Image)
6.448 us

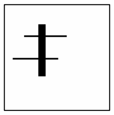
28-Aug-08 Binary Morphology 36

Exercises with Opening and Closing

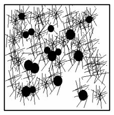
a) Use image ballbar.jl
Split ball from bar



b) Use image bars.jl
Split in one vertical bar and two horizontal bars



c) Use image ballstripes.jl
Filter the balls from the image



Answers:

- **ballbar.jls**
- **bars.jls**
- **ballstripes.jls**

Hit and Miss (*)

HitAndMiss (src, dest, hitMask, missMask)

The destination image has pixels set to the object value (= 1) at positions where all objects in the hitmask are contained in the source image and none of the object pixels in the missmask are contained in the source image.

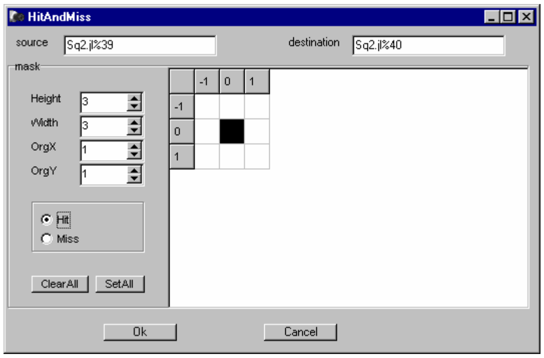
Usage: to find specific points of blobs like corners or borders.

Demonstration Hit and Miss, find right border (*)

- **Example find right border**
  - **Open file sq2.jl**
  - **Apply HitAndMiss with hitmask: 0 0 0 and missmask: 0 0 1**

0 1 0 0 0 1
0 0 0 0 0 1

Hit mask (*)



source Sq2.jl%39

destination Sq2.jl%40

Height 3

Width 3

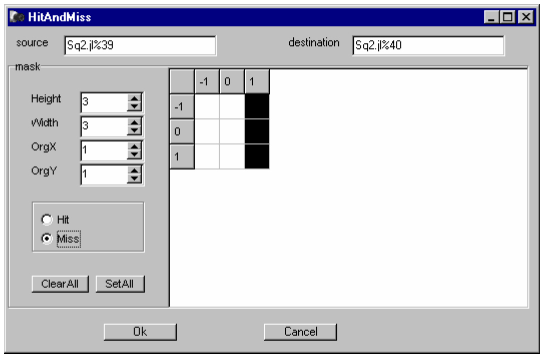
OrgX 1

OrgY 1

● Hit
○ Miss

	-1	0	1
-1			
0		■	
1			

Miss mask (*)



HitAndMiss

source: Sq2.jl%39 destination: Sq2.jl%40

mask

Height: 3

Width: 3

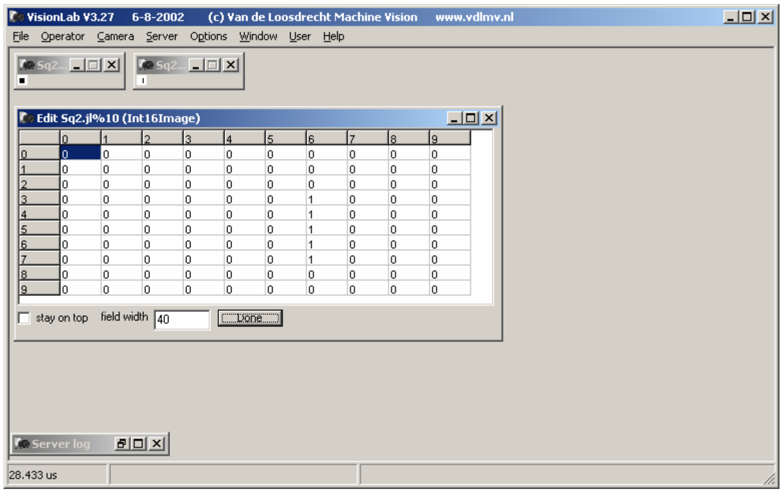
OrgX: 1

OrgY: 1

○ Hit
● Miss

	-1	0	1
-1			■
0			■
1			■

Demonstration Hit and Miss, find right border (*)



VisionLab V3.27 6-8-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File
Operator
Camera
Server
Options
Window
User
Help
Sq2...
Sq2...

Edit Sq2.jl%10 (Int16Image)

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	1	0	0	0
4	0	0	0	0	0	0	1	0	0	0
5	0	0	0	0	0	0	1	0	0	0
6	0	0	0	0	0	0	1	0	0	0
7	0	0	0	0	0	0	1	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

□ stay on top field width: 40

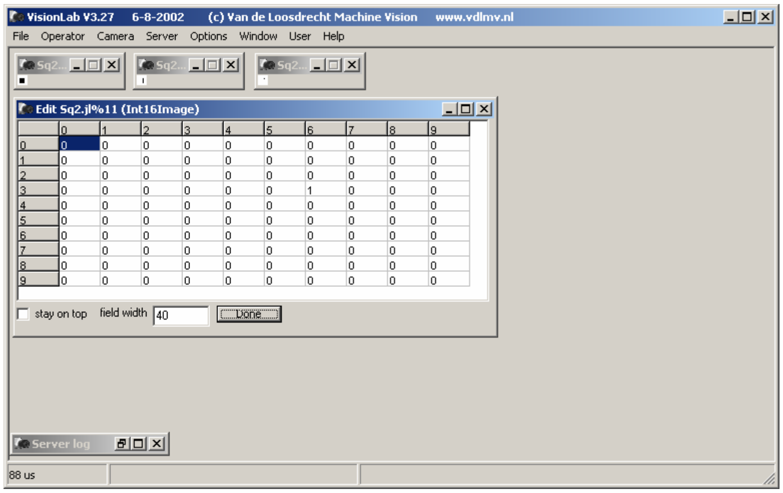
28.433 us

Demonstration Hit and Miss, find right top corner (*)

- Example find right top corner
  - Open file sq2.jl
  - Apply HitAndMiss with hitmask: 0 0 0 and missmask: 0 1 1

```
                                  1 1 0                0 0 1
                                  0 1 0                0 0 0
```

Demonstration Hit and Miss, find right top corner (*)



Edit Sq2.jl%11 (Int16Image)

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	1	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

☐ stay on top field width 40 Done

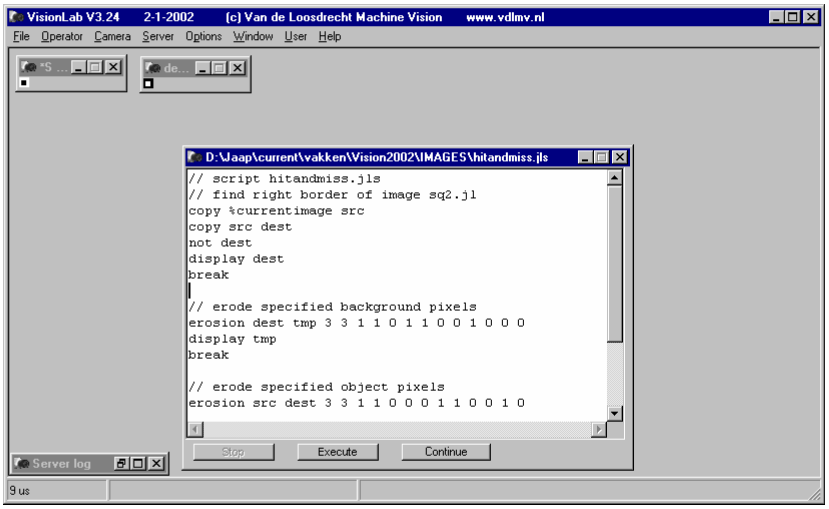
Hit and Miss (*)

Algorithm:

- **dest = src**
- **Not (dest)**
- **Erosion (dest, tmp, missMask) // 'dilate' background**
- **Erosion (src, dest, hitMask) // erode objects**
- **dest &= tmp;**

- **see script hitandmiss.jls**

Not (dest) (*)

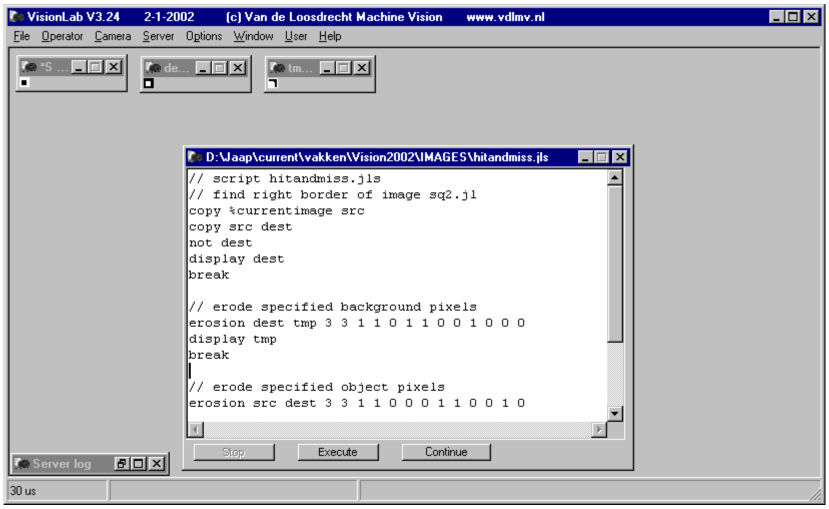

VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
*S ...
de...
D:\Jaap\current\vakken\Vision2002\IMAGES\hitandmiss.jls
Stop
Execute
Continue
Server log
9 us

```
// script hitandmiss.jls
// find right border of image sq2.jl
copy %currentimage src
copy src dest
not dest
display dest
break

// erode specified background pixels
erosion dest tmp 3 3 1 1 0 1 1 0 0 1 0 0 0
display tmp
break

// erode specified object pixels
erosion src dest 3 3 1 1 0 0 0 1 1 0 0 1 0
```

Erosion (dest, tmp, missMask) (*)


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
*S ...
de...
tm...
D:\Jaap\current\vakken\Vision2002\IMAGES\hitandmiss.jls
Stop
Execute
Continue
Server log
30 us

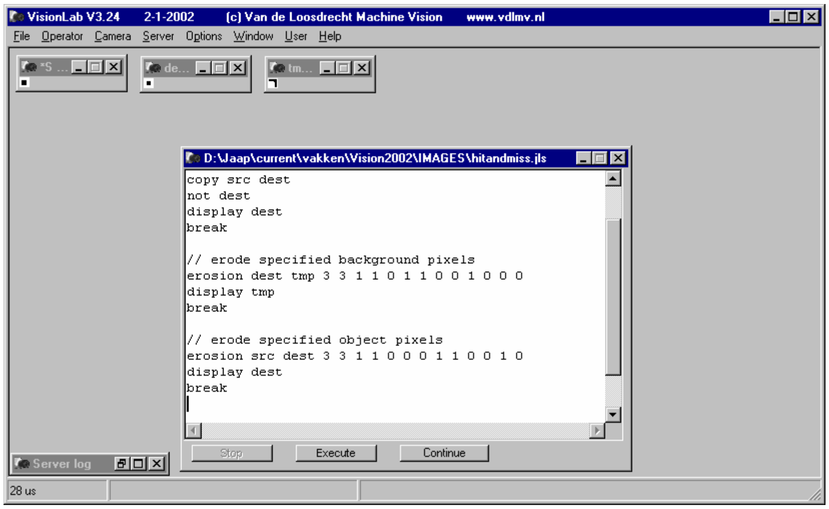
```
// script hitandmiss.jls
// find right border of image sq2.jl
copy %currentimage src
copy src dest
not dest
display dest
break

// erode specified background pixels
erosion dest tmp 3 3 1 1 0 1 1 0 0 1 0 0 0
display tmp
break

// erode specified object pixels
erosion src dest 3 3 1 1 0 0 0 1 1 0 0 1 0
```

28-Aug-08 Binary Morphology 47

Erosion (src, dest, hitMask) (*)


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
*S ...
de...
tm...
D:\Jaap\current\vakken\Vision2002\IMAGES\hitandmiss.jls
Stop
Execute
Continue
Server log
28 us

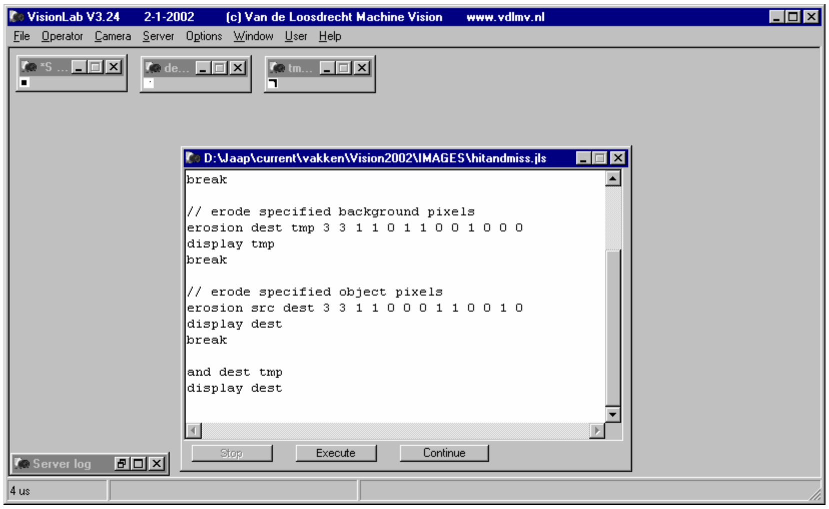
```
copy src dest
not dest
display dest
break

// erode specified background pixels
erosion dest tmp 3 3 1 1 0 1 1 0 0 1 0 0 0
display tmp
break

// erode specified object pixels
erosion src dest 3 3 1 1 0 0 0 1 1 0 0 1 0
display dest
break
```

28-Aug-08 Binary Morphology 48

dest &= tmp; (*)


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
*S ...
de...
tm...
D:\Jaap\current\vakken\Vision2002\IMAGES\hitandmiss.jls
Stop
Execute
Continue
Server log
4 us

```
break

// erode specified background pixels
erosion dest tmp 3 3 1 1 0 1 1 0 0 1 0 0 0
display tmp
break

// erode specified object pixels
erosion src dest 3 3 1 1 0 0 0 1 1 0 0 1 0
display dest
break

and dest tmp
display dest
```

Thinning (*)

Thinning (src, dest, hitMask, missMask)

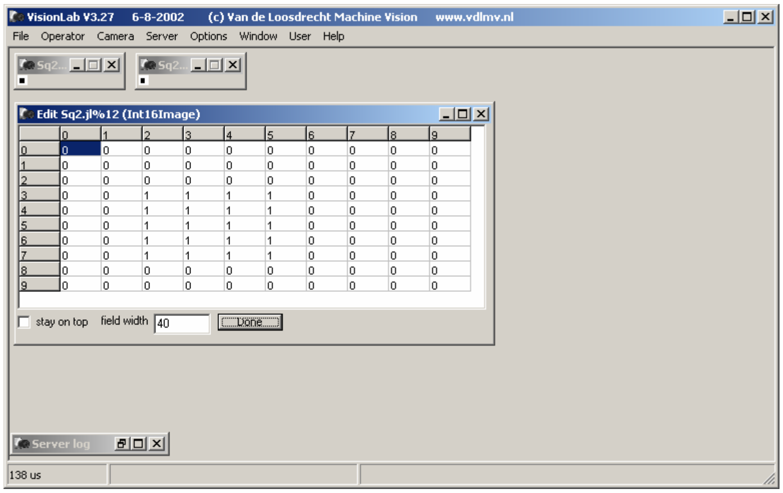
The destination image is the same as the source image but object pixels which are covered by the hitmask and not covered by the missmask are excluded from the destination image.
In order to perform meaningful, the origin of the hitmask should be *included* in the hitmask and the origin of the missmask should be *excluded* from the missmask.

Usage: at selective points peeling the blobs.

Demonstration Thinning, remove right border (*)

- Open image sq2.jl
- Analyse image with Edit
- Apply Thinning with hitmask: 0 0 0 and missmask: 0 0 1
  0 1 0 0 0 1
  0 0 0 0 0 1
- Analyse result with Edit; right border has been removed

Demonstration Thinning, remove right border (*)


VisionLab V3.27 6-8-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File
Operator
Camera
Server
Options
Window
User
Help
Sq2...
Sq2...
Edit Sq2.jl%12 (Int16Image)
stay on top
Done
Server log

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	1	1	1	1	0	0	0	0
4	0	0	1	1	1	1	0	0	0	0
5	0	0	1	1	1	1	0	0	0	0
6	0	0	1	1	1	1	0	0	0	0
7	0	0	1	1	1	1	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

☐ stay on top field width 40

138 us

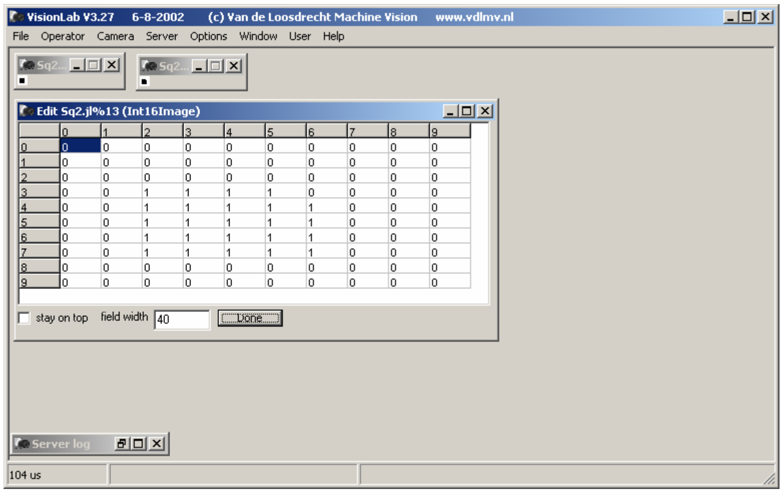
Demonstration Thinning, remove right top (*)

- Open image sq2.jl
- Analyse image with Edit
- Apply Thinning with hitmask: 0 0 0 and missmask: 0 1 1

```
                               1 1 0                  0 0 1
                               0 1 0                  0 0 0
```

- Analyse result with Edit; right top has been removed

Demonstration Thinning, remove right top (*)



VisionLab V3.27 6-8-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Sq2...
Sq2...
Edit Sq2.jl%13 (Int16Image)

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	1	1	1	1	0	0	0	0
4	0	0	1	1	1	1	1	0	0	0
5	0	0	1	1	1	1	1	0	0	0
6	0	0	1	1	1	1	1	0	0	0
7	0	0	1	1	1	1	1	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

☐ stay on top field width 40

Thinning (*)

Algorithm:

- **HitAndMiss (src, tmp, hitMask, missMask)**
- **dest = src**
- **dest -= tmp (exor is also possible)**

Thickening (*)

Thickening (src, dest, hitMask, missMask)

The destination image is the same as the source image but at special positions pixels are added.

These positions are where all objects in the hitmask are contained in the source image and none of the object pixels in the missmask are contained in the source image.

In order to perform meaningful, the origin of the hitmask should be *excluded* from the hitmask and the origin of the missmask should be *included* in the missmask.
Note: only the origin pixel is added.

Usage: add at selective points a layer to the blobs

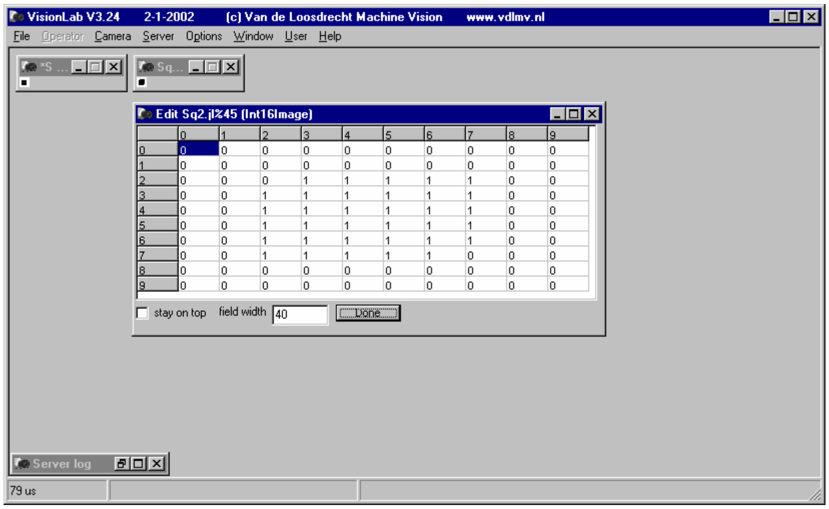
Demonstration Thickening (*)

- **Open image sq2.jl**
- **Analyse image with Edit**
- **Apply Thickening with hitmask: 0 0 0 and missmask: 0 0 0**
  0 0 0 0 1 0
  1 0 0 0 0 0

  must be present

  is not present and is added
- **Analyse result with Edit; a layer is added at the top and the right side, note top left and bottom right.**

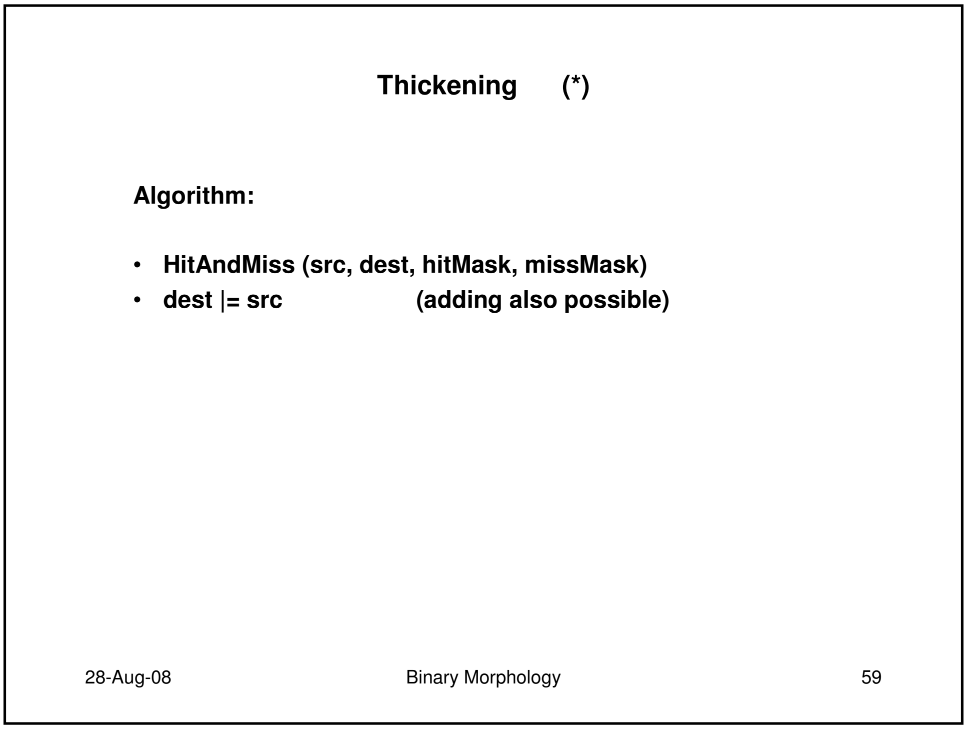
Demonstration Thickening (*)


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
*S ...
Sq...
Server log
79 us

Edit Sq2.jl%45 (Int16Image)

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	1	1	1	1	1	0	0
3	0	0	1	1	1	1	1	1	0	0
4	0	0	1	1	1	1	1	1	0	0
5	0	0	1	1	1	1	1	1	0	0
6	0	0	1	1	1	1	1	1	0	0
7	0	0	1	1	1	1	1	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

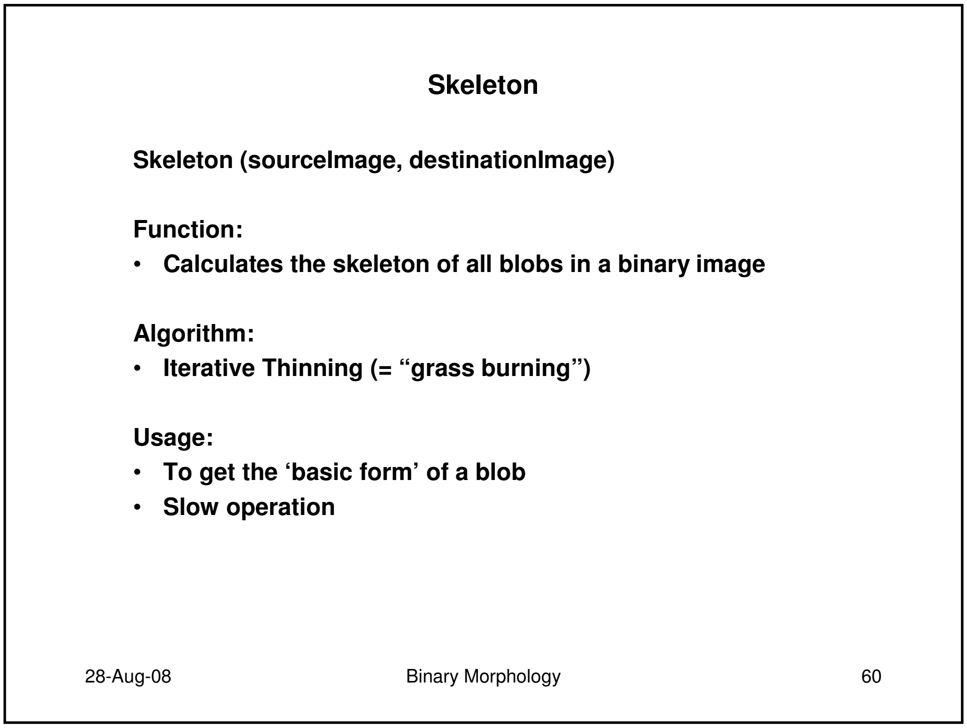
□ stay on top field width 40 Done



Thickening (*)

Algorithm:

- **HitAndMiss (src, dest, hitMask, missMask)**
- **dest |= src (adding also possible)**



Skeleton

Skeleton (sourceImage, destinationImage)

Function:

- **Calculates the skeleton of all blobs in a binary image**

Algorithm:

- **Iterative Thinning (= “grass burning”)**

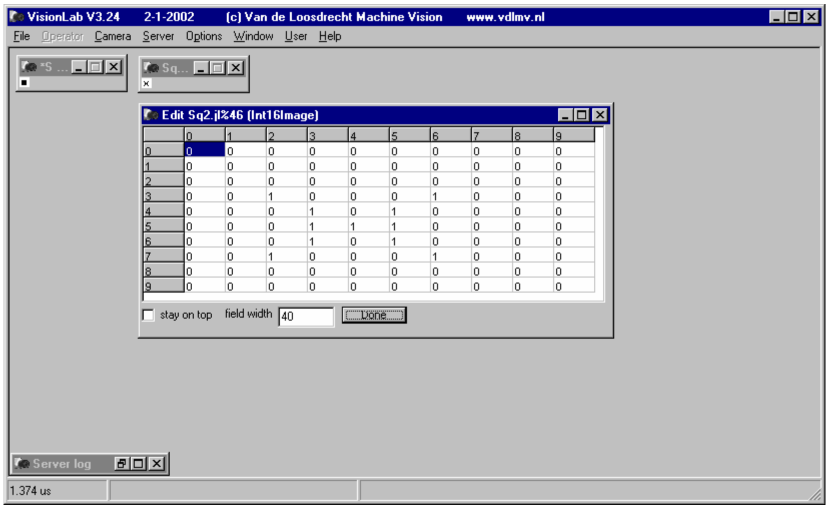
Usage:

- **To get the ‘basic form’ of a blob**
- **Slow operation**

Demonstration Skeleton sq.jl

- Open image sq.jl
- Apply Skeleton
- Analyse result with Edit

Demonstration Skeleton sq.jl


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
*S ...
Sq...
Edit Sq2.jl%46 (Int16Image)
Server log

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	1	0	0	0	1	0	0	0
4	0	0	0	1	0	1	0	0	0	0
5	0	0	0	1	1	1	0	0	0	0
6	0	0	0	1	0	1	0	0	0	0
7	0	0	1	0	0	0	1	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

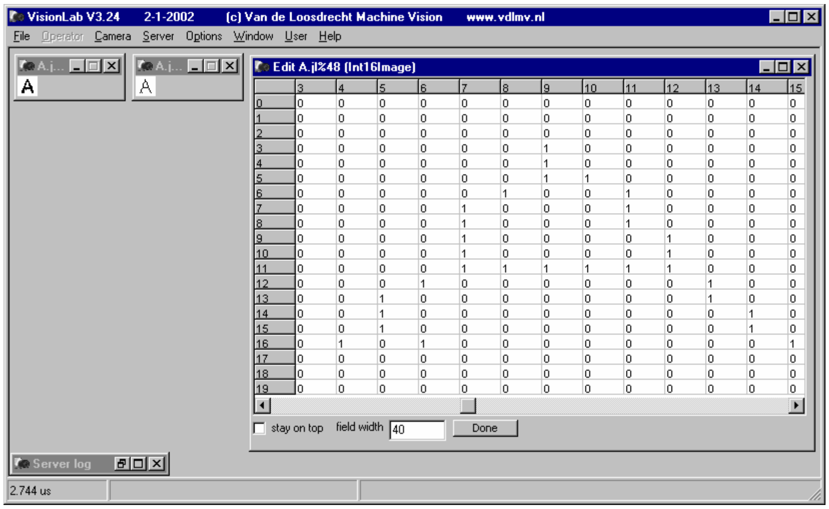
☐ stay on top field width 40 Done

1.374 us

Demonstration Skeleton A.jl

- Open image A.jl
- Apply Skeleton
- Analyse result with Edit

Demonstration Skeleton A.jl


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
A.j...
A.j...
A
A
Server log

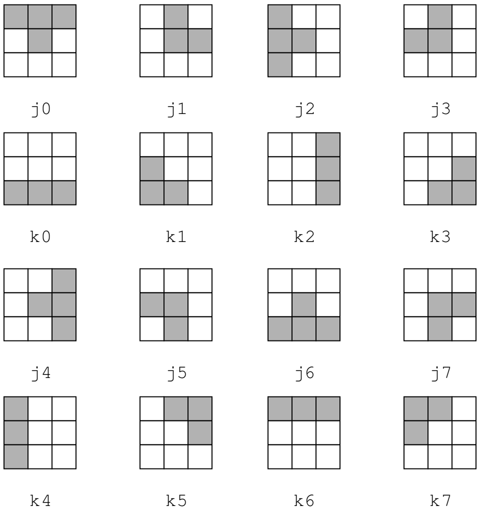
Edit A.jl%48 (Int16Image)

	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	1	0	0	0	0	0	0
4	0	0	0	0	0	0	1	0	0	0	0	0	0
5	0	0	0	0	0	0	1	1	0	0	0	0	0
6	0	0	0	0	0	1	0	0	1	0	0	0	0
7	0	0	0	0	1	0	0	0	1	0	0	0	0
8	0	0	0	0	1	0	0	0	1	0	0	0	0
9	0	0	0	0	1	0	0	0	0	1	0	0	0
10	0	0	0	0	1	0	0	0	0	1	0	0	0
11	0	0	0	0	1	1	1	1	1	1	0	0	0
12	0	0	0	1	0	0	0	0	0	0	1	0	0
13	0	0	1	0	0	0	0	0	0	0	1	0	0
14	0	0	1	0	0	0	0	0	0	0	0	1	0
15	0	0	1	0	0	0	0	0	0	0	0	1	0
16	0	1	0	1	0	0	0	0	0	0	0	0	1
17	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0

□ stay on top field width 40 Done

2.744 us

Skeleton implementation masks (*)



Skeleton algorithm (*)

```
dest = src;
do {
  prevImage = dest;
  for (int i = 0; i < 8; i += 2) {
              Thinning (dest, tmp, j[i], k[i]);
              Thinning (tmp, dest, j[i+1], k[i+1]);
  }
} while (prevImage != dest);
```

Note:

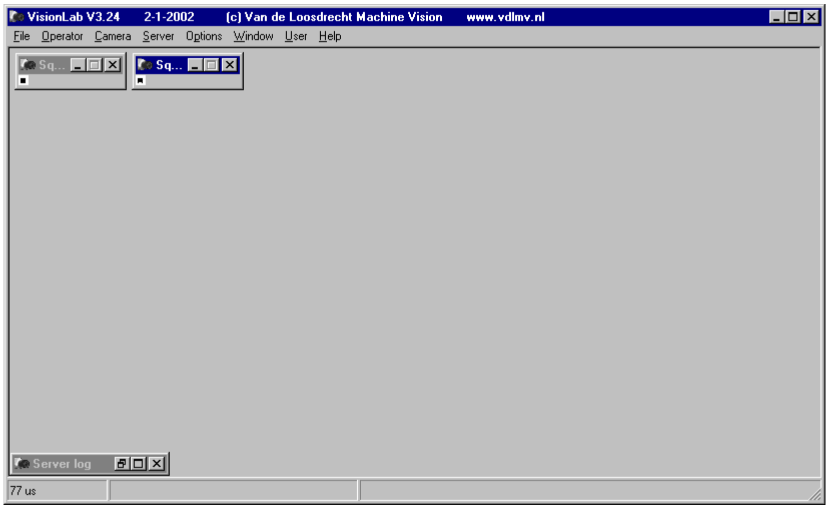
- **This is a very slow implementation**
- **Alternatives can be based on distance transform**

Demonstration Skeleton algorithm (*)

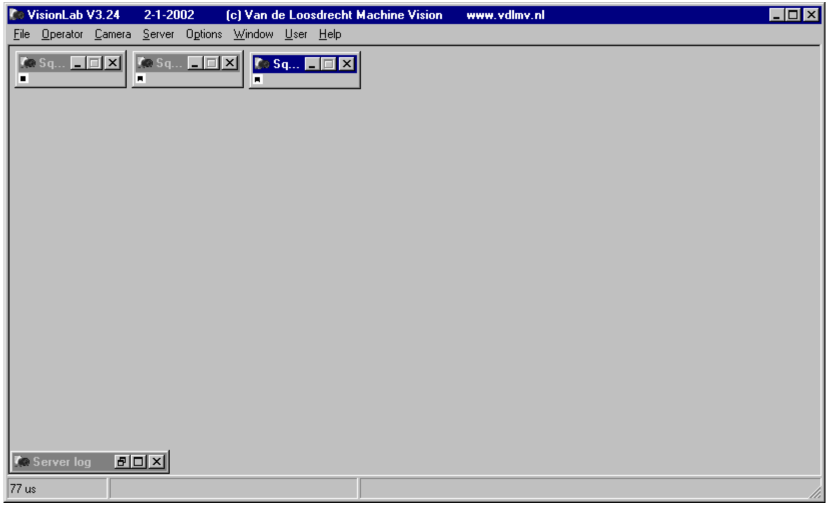
- **Open image sq2.jl**
- **Thinning with masks j0 and k0,**
  eats middle of bottom border, but excludes the corners
- **Thinning result with masks j1 and k1,**
  tries to eat left bottom corner, unless it is an isolated corner

- **To see that Thinning with masks j1 and k1,**
  tries to eat left bottom corner, unless it is an isolated corner,
  do this operation on sq2.jl

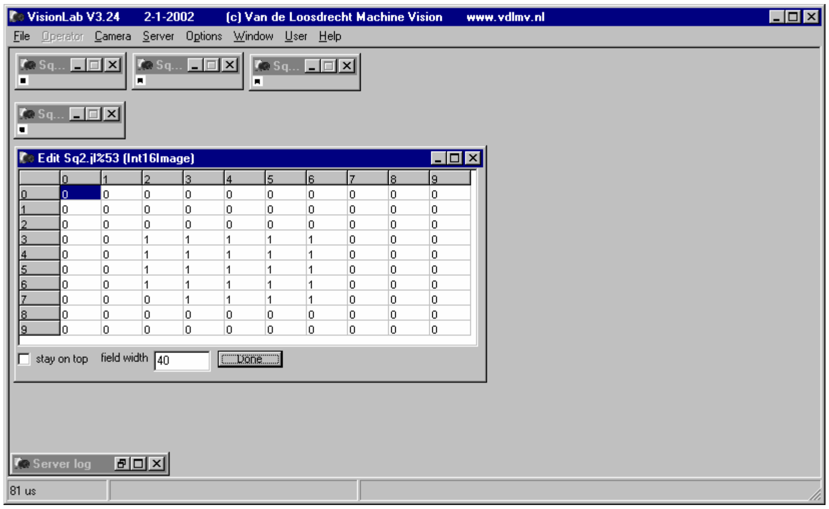
Thinning with masks j0 and k0, (*) eats middle of bottom border, but excludes the corners


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File
Operator
Camera
Server
Options
Window
User
Help
Sq...
Sq...
Server log
77 us

Thinning result with masks j1 and k1, (*)
tries to eat left bottom corner, unless it is an isolated corner


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Sq...
Sq...
Sq...
Server log
77 us

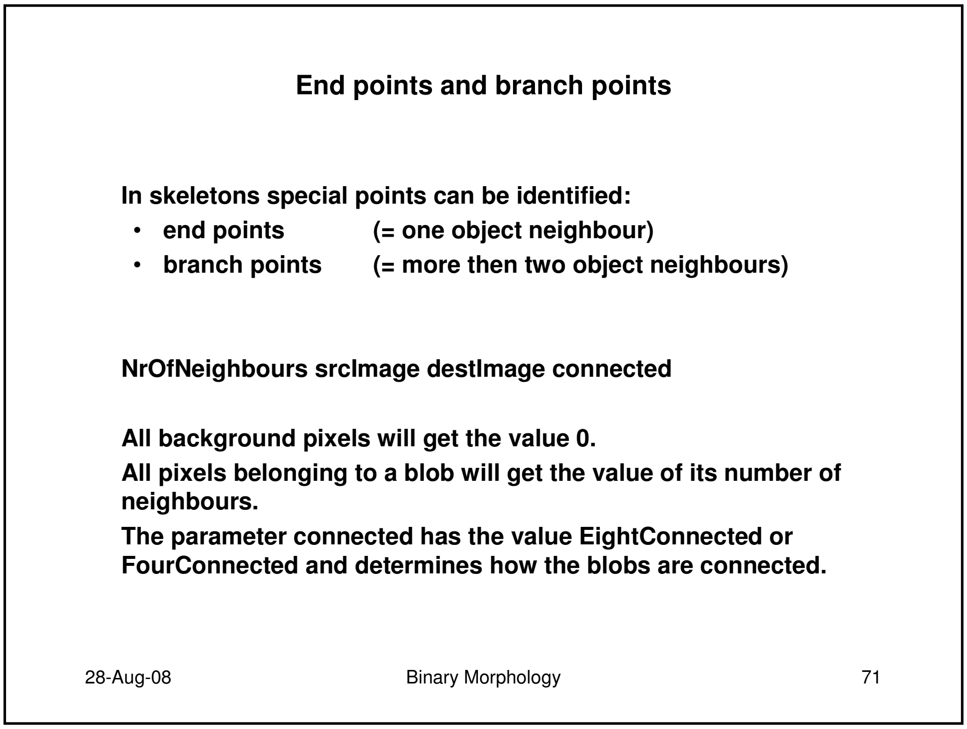
Thinning with masks j1 and k1 on sq.jl, (*)
tries to eat left bottom corner, unless it is an isolated corner


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
Sq...
Sq...
Sq...
Sq...
Server log
81 us

Edit Sq2.jl%53 (Int16Image)

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	1	1	1	1	1	0	0	0
4	0	0	1	1	1	1	1	0	0	0
5	0	0	1	1	1	1	1	0	0	0
6	0	0	1	1	1	1	1	0	0	0
7	0	0	0	1	1	1	1	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0

□ stay on top field width 40 Done



End points and branch points

In skeletons special points can be identified:

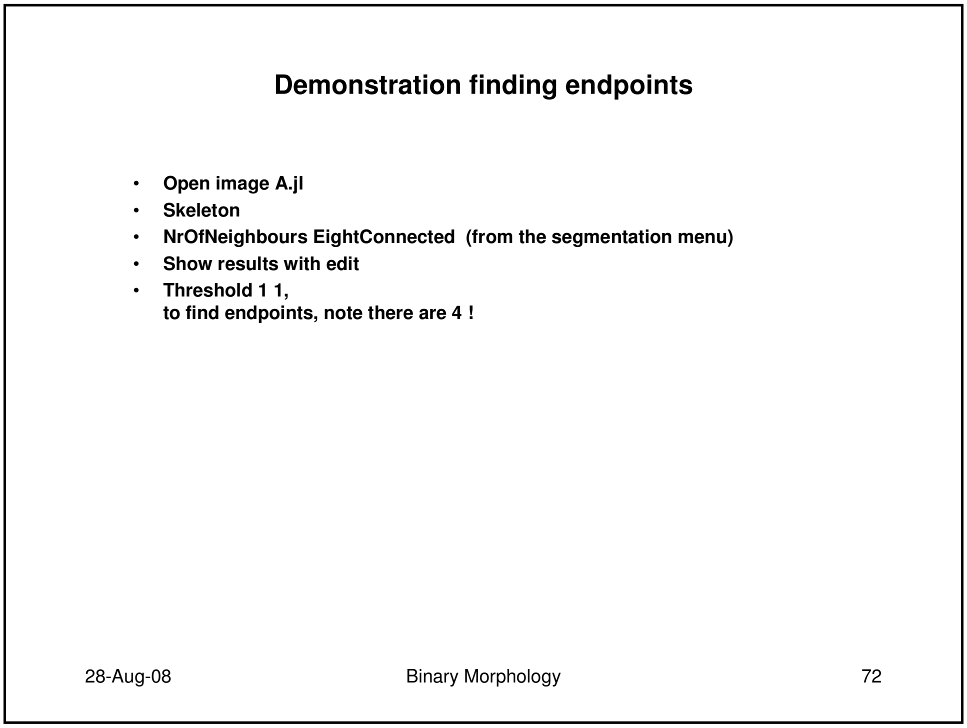
- **end points (= one object neighbour)**
- **branch points (= more then two object neighbours)**

NrOfNeighbours srcImage destImage connected

All background pixels will get the value 0.

All pixels belonging to a blob will get the value of its number of neighbours.

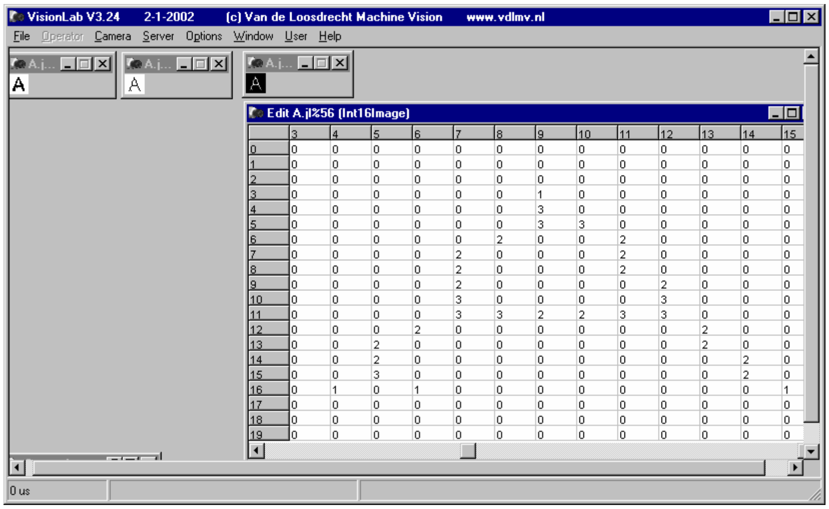
The parameter connected has the value EightConnected or FourConnected and determines how the blobs are connected.



Demonstration finding endpoints

- **Open image A.jl**
- **Skeleton**
- **NrOfNeighbours EightConnected (from the segmentation menu)**
- **Show results with edit**
- **Threshold 1 1,**
  to find endpoints, note there are 4 !

NrOfNeighbours

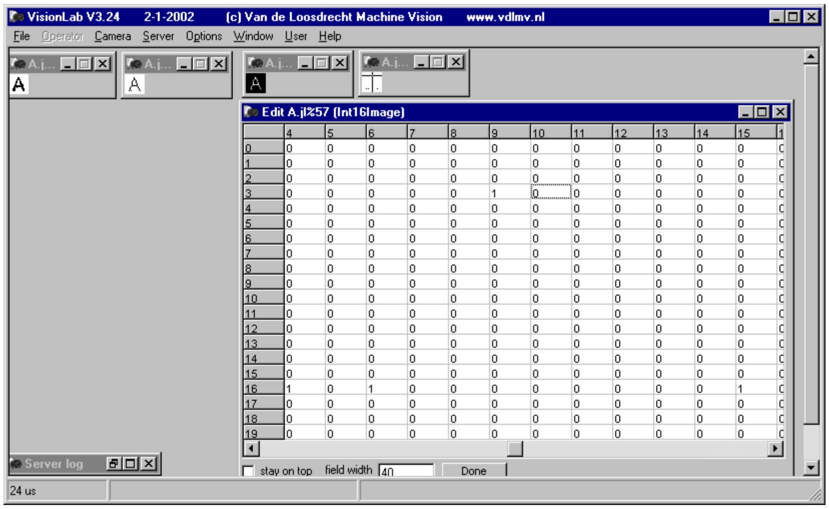

VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
A.j...
A.j...
A.j...
0 us

Edit A.jl%56 (Int16Image)

	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	1	0	0	0	0	0	0
4	0	0	0	0	0	0	3	0	0	0	0	0	0
5	0	0	0	0	0	0	3	3	0	0	0	0	0
6	0	0	0	0	0	2	0	0	2	0	0	0	0
7	0	0	0	0	2	0	0	0	2	0	0	0	0
8	0	0	0	0	2	0	0	0	2	0	0	0	0
9	0	0	0	0	2	0	0	0	0	2	0	0	0
10	0	0	0	0	3	0	0	0	0	3	0	0	0
11	0	0	0	0	3	3	2	2	3	3	0	0	0
12	0	0	0	2	0	0	0	0	0	0	2	0	0
13	0	0	2	0	0	0	0	0	0	0	2	0	0
14	0	0	2	0	0	0	0	0	0	0	0	2	0
15	0	0	3	0	0	0	0	0	0	0	0	2	0
16	0	1	0	1	0	0	0	0	0	0	0	0	1
17	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0

28-Aug-08 Binary Morphology 73

Threshold 1 1, to find 4 endpoints


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
A.j...
A.j...
A.j...
A.j...
Server log
□ stay on top
field width 40
Done
24 us

Edit A.jl%57 (Int16Image)

	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0
16	1	0	1	0	0	0	0	0	0	0	0	1
17	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0

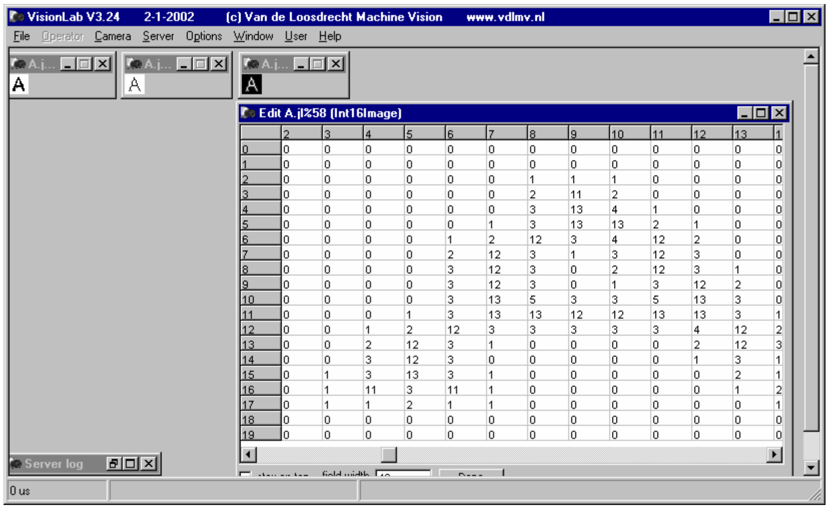
28-Aug-08 Binary Morphology 74

Implementation of NrOfNeigbours (*)

For EightConnected objects:

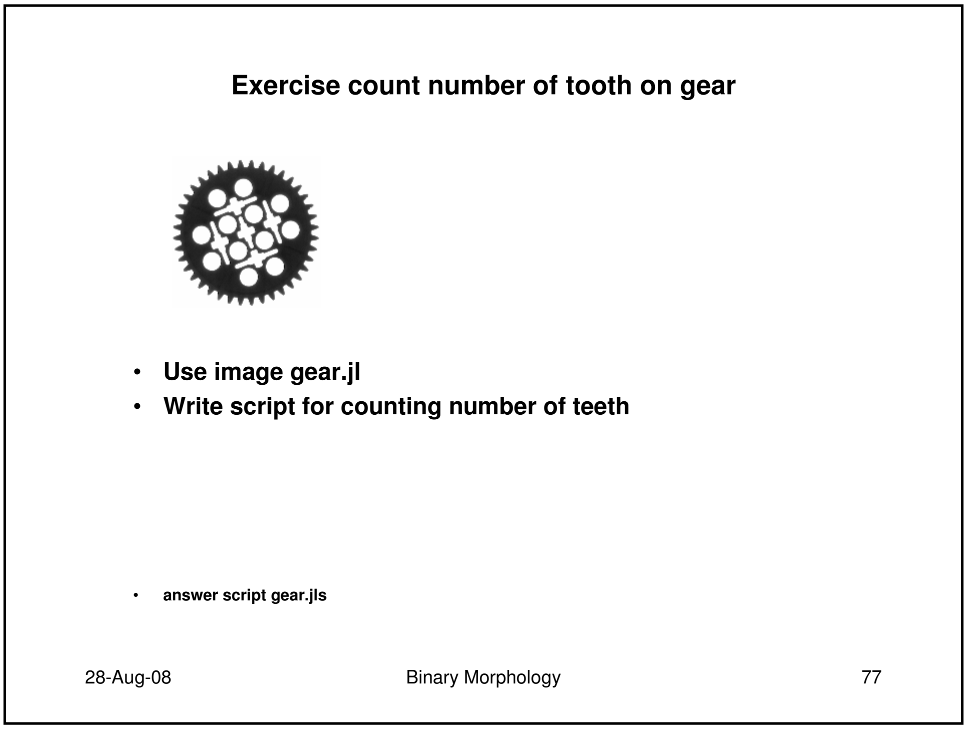
- Convolution with mask $\begin{matrix} 1 & 1 & 1 \\ 1 & 10 & 1 \\ 1 & 1 & 1 \end{matrix}$
  - p < 10: background pixel with p number of objects as neighbour
  - p > 10: object pixel with p - 10 number of objects as neighbour
- Subtract 10 from all pixels
- Setselectedtovalue -10 -1 0

Convolution on skeleton (*)


VisionLab V3.24 2-1-2002 (c) Van de Loosdrecht Machine Vision www.vdlmv.nl
File Operator Camera Server Options Window User Help
A.j...
A
A.j...
A
A.j...
A
Server log
0 us

Edit A.jl%58 (Int16Image)

	2	3	4	5	6	7	8	9	10	11	12	13	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	1	1	1	0	0	0	0
3	0	0	0	0	0	0	2	11	2	0	0	0	0
4	0	0	0	0	0	0	3	13	4	1	0	0	0
5	0	0	0	0	0	1	3	13	13	2	1	0	0
6	0	0	0	0	1	2	12	3	4	12	2	0	0
7	0	0	0	0	2	12	3	1	3	12	3	0	0
8	0	0	0	0	3	12	3	0	2	12	3	1	0
9	0	0	0	0	3	12	3	0	1	3	12	2	0
10	0	0	0	0	3	13	5	3	3	5	13	3	0
11	0	0	0	1	3	13	13	12	12	13	13	3	1
12	0	0	1	2	12	3	3	3	3	3	4	12	2
13	0	0	2	12	3	1	0	0	0	0	2	12	3
14	0	0	3	12	3	0	0	0	0	0	1	3	1
15	0	1	3	13	3	1	0	0	0	0	0	2	1
16	0	1	11	3	11	1	0	0	0	0	0	1	2
17	0	1	1	2	1	1	0	0	0	0	0	0	1
18	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0



Exercise count number of tooth on gear

- Use image gear.jl
- Write script for counting number of teeth

- answer script gear.jls