Seminar: Medical Image Processing

A robust approach for automatic detection and segmentation of cracks in underground pipeline images

Tim Niemueller <tim@niemueller.de>

Supervisor: Benedikt Fischer
Institut für medizinische Informatik, RWTH Aachen

July 13th, 2006

Roadmap

Discussed papers

- Shivprakash Iyer and Sunil K. Sinha: A robust approach for automatic detection and segmentation of cracks in underground pipeline images. *Image and Vision Computing*, 23:921-933, 2005.

Discussed papers

- Shivprakash Iyer and Sunil K. Sinha: A robust approach for automatic detection and segmentation of cracks in underground pipeline images. *Image and Vision Computing*, 23:921-933, 2005.
- Frederic Zana and Jean-Claude Klein: Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation. *IEEE Transactions on Image Processing*, 10(7):1010-1019, July 2001.

The situation

- Communal sewer networks often one of the biggest infrastructures in an industrialized country (USA: approx. 1 million miles)

The situation

- Communal sewer networks often one of the biggest infrastructures in an industrialized country (USA: approx. 1 million miles)
- Networks built 50-60 years ago

The situation

- Communal sewer networks often one of the biggest infrastructures in an industrialized country (USA: approx. 1 million miles)
- Networks built 50-60 years ago
- Networks age and deteriorate until they fail

The situation

- Communal sewer networks often one of the biggest infrastructures in an industrialized country (USA: approx. 1 million miles)
- Networks built 50-60 years ago
- Networks age and deteriorate until they fail
- Pipes are in general too small for humans

The situation

- Communal sewer networks often one of the biggest infrastructures in an industrialized country (USA: approx. 1 million miles)
- Networks built 50-60 years ago
- Networks age and deteriorate until they fail
- Pipes are in general too small for humans
- Images can be taken via installed camera or by semi-mobile robots

The situation

- Communal sewer networks often one of the biggest infrastructures in an industrialized country (USA: approx. 1 million miles)
- Networks built 50-60 years ago
- Networks age and deteriorate until they fail
- Pipes are in general too small for humans
- Images can be taken via installed camera or by semi-mobile robots

Large underground sewer networks need continuous checks.

The problem

- Continuous check needed to guarantee fitness

The problem

- Continuous check needed to guarantee fitness
- Currently these checks are done manually

The problem

- Continuous check needed to guarantee fitness
- Currently these checks are done manually
- Checks highly dependent on experience, concentration and skill level of operator

The problem

- Continuous check needed to guarantee fitness
- Currently these checks are done manually
- Checks highly dependent on experience, concentration and skill level of operator
- Human operators: subjectivity, fatigue, high costs

The problem

- Continuous check needed to guarantee fitness
- Currently these checks are done manually
- Checks highly dependent on experience, concentration and skill level of operator
- Human operators: subjectivity, fatigue, high costs

Reliable *automated defect detection* and classification system desirable to compensate these problems

What are cracks?

- Large linear portions

What are cracks?

- Large linear portions
- Branch like a tree

What are cracks?

- Large linear portions
- Branch like a tree
- Intensity distribution of a crack feature cross-section looks like a specific gaussian curve

What are cracks?

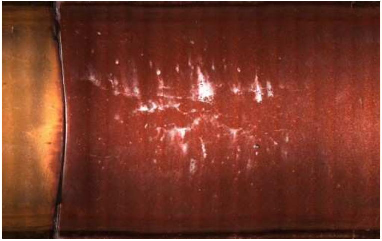
- Large linear portions
- Branch like a tree
- Intensity distribution of a crack feature cross-section looks like a specific gaussian curve
- More or less constant width

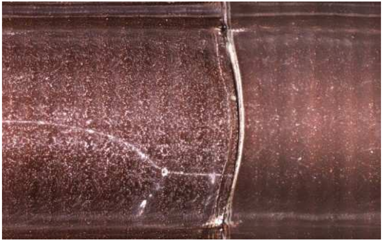
What are cracks?

- Large linear portions
- Branch like a tree
- Intensity distribution of a crack feature cross-section looks like a specific gaussian curve
- More or less constant width

- Retinal vessels: similar features ⇒ similar method works to segment vessels

Examples (cracks and retinal vessels)





Examples (cracks and retinal vessels)

Examples (cracks and retinal vessels)

Processing pipeline structure

- Usage of mathematical morphology (MM) and linear filters (LF)

Processing pipeline structure

- Usage of mathematical morphology (MM) and linear filters (LF)
- Results in binary crack map

Processing pipeline structure

- Usage of mathematical morphology (MM) and linear filters (LF)
- Results in binary crack map
- Basic 3-step processing pipeline

Processing pipeline structure

- Usage of mathematical morphology (MM) and linear filters (LF)
- Results in binary crack map
- Basic 3-step processing pipeline

1. Preprocessing (contrast enhancement)

Processing pipeline structure

- Usage of mathematical morphology (MM) and linear filters (LF)
- Results in binary crack map
- Basic 3-step processing pipeline

1. Preprocessing (contrast enhancement)
2. Enhancement of cracks (MM and LF)

Processing pipeline structure

- Usage of mathematical morphology (MM) and linear filters (LF)
- Results in binary crack map
- Basic 3-step processing pipeline

1. Preprocessing (contrast enhancement)
2. Enhancement of cracks (MM and LF)
3. Segmentation of cracks (MM alternating filters)

1 Introduction

2 Morphology
- What is mathematical morphology?
- Morphology operations
- Specific parameters for crack detection

3 Linear filters

4 Detection

5 Evaluation

6 Summary

Introduction

- Mathematical morphology (MM) developed by Matheron and Serra at the Ecole des Mines in Paris

Introduction

- Mathematical morphology (MM) developed by Matheron and Serra at the Ecole des Mines in Paris
- Extract features based on a priori knowledge about object geometry

Introduction

- Mathematical morphology (MM) developed by Matheron and Serra at the Ecole des Mines in Paris
- Extract features based on a priori knowledge about object geometry
- Set-theoretic method providing a quantitative description of geometric structures

Introduction

- Mathematical morphology (MM) developed by Matheron and Serra at the Ecole des Mines in Paris
- Extract features based on a priori knowledge about object geometry
- Set-theoretic method providing a quantitative description of geometric structures
- Based on expanding and shrinking operations with regard to a given structuring element (knowledge about object)

Introduction

- Mathematical morphology (MM) developed by Matheron and Serra at the Ecole des Mines in Paris
- Extract features based on a priori knowledge about object geometry
- Set-theoretic method providing a quantitative description of geometric structures
- Based on expanding and shrinking operations with regard to a given structuring element (knowledge about object)
- Originally for B/W images, extended for gray images (interesting case here)

Definitions

Images are defined as a function mapping from points to intensity values (here: grayscale, $I_{min} = 0$ and $I_{max} = 255$):

$$F : \mathbb{Z}^2 \mapsto [I_{min}, I_{max}]$$

Definitions

Images are defined as a function mapping from points to intensity values (here: grayscale, $I_{min} = 0$ and $I_{max} = 255$):

$$F : \mathbb{Z}^2 \mapsto [I_{min}, I_{max}]$$

Binary *structuring elements* (SE) are defined as a function:

$$B : \mathbb{Z}^2 \mapsto [0, 1]$$

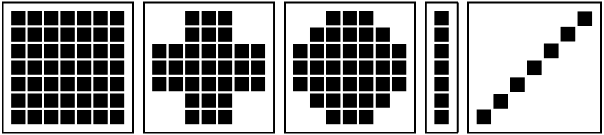
Definitions

Images are defined as a function mapping from points to intensity values (here: grayscale, $I_{min} = 0$ and $I_{max} = 255$):

$$F : \mathbb{Z}^2 \mapsto [I_{min}, I_{max}]$$

Binary *structuring elements* (SE) are defined as a function:

$$B : \mathbb{Z}^2 \mapsto [0, 1]$$



Important notation specialties for crack detection

General MM:

- Foreground: white
- Background: black

Crack detection MM:

- Foreground: black
- Background: white

Important notation specialties for crack detection

General MM:

- Foreground: white
- Background: black

Crack detection MM:

- Foreground: black
- Background: white

Some items change meaning:

- Hole
- Object

- Object
- Hole

Important notation specialties for crack detection

General MM:

- Foreground: white
- Background: black

Crack detection MM:

- Foreground: black
- Background: white

Some items change meaning:

General MM	Crack detection MM
Hole	Object
Object	Hole

Some operations change meaning:

General MM	Crack detection MM
Expanding	Shrinking
Shrinking	Expanding

Dilation

$$\delta_B^e(F)(P_0) = \max_{P \in P_0 \cup e \cdot B(P_0)}(F(P))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image
P_0	Point in image (repeat for every point)

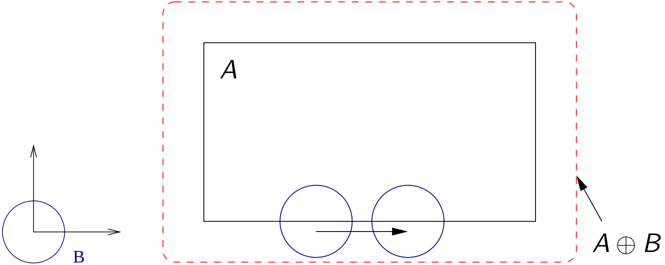
- Basic *expanding* operation.

Dilation

$$\delta_B^e(F)(P_0) = \max_{P \in P_0 \cup e \cdot B(P_0)}(F(P))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image
P_0	Point in image (repeat for every point)

- Basic *expanding* operation.

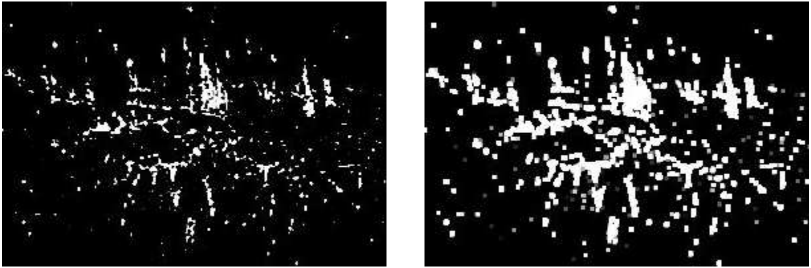


Dilation

$$\delta_B^e(F)(P_0) = \max_{P \in P_0 \cup e \cdot B(P_0)}(F(P))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image
P_0	Point in image (repeat for every point)

- Basic *expanding* operation.



(general, black background, white foreground)

Dilation

$$\delta_B^e(F)(P_0) = \max_{P \in P_0 \cup e \cdot B(P_0)}(F(P))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image
P_0	Point in image (repeat for every point)

- Basic *expanding* operation.

(crack detection, white background, black foreground)

Erosion

$$\varepsilon_B^e(F)(P_0) = \min_{P \in P_0 \cup e \cdot B(P_0)}(F(P))$$

- Basic *shrinking* operation.

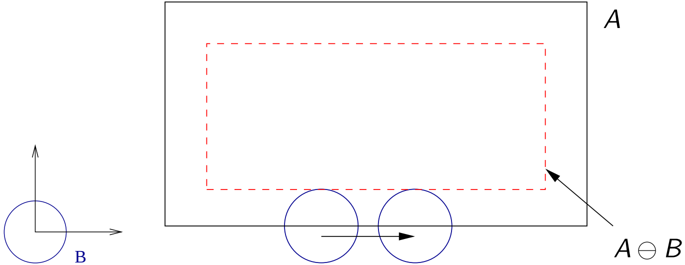
B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image
P_0	Point in image (repeat for every point)

Erosion

$$\varepsilon_B^e(F)(P_0) = \min_{P \in P_0 \cup e \cdot B(P_0)}(F(P))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image
P_0	Point in image (repeat for every point)

- Basic *shrinking* operation.

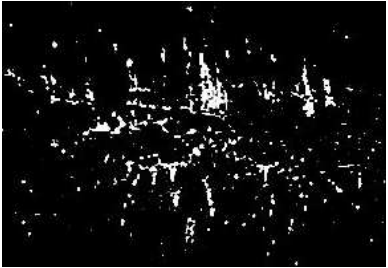


Erosion

$$\varepsilon_B^e(F)(P_0) = \min_{P \in P_0 \cup e \cdot B(P_0)}(F(P))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image
P_0	Point in image (repeat for every point)

- Basic *shrinking* operation.



(general, black background, white foreground)

Erosion

$$\varepsilon_B^e(F)(P_0) = \min_{P \in P_0 \cup e \cdot B(P_0)}(F(P))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image
P_0	Point in image (repeat for every point)

- Basic *shrinking* operation.

(crack detection, white background, black foreground)

Opening

$$\gamma_B^e(F) = \delta_B^e(\varepsilon_B^e(F))$$

- Dilation of the erosion

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

Opening

$$\gamma_B^e(F) = \delta_B^e(\varepsilon_B^e(F))$$

- Dilation of the erosion
- Removes small objects

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

Opening

$$\gamma_B^e(F) = \delta_B^e(\varepsilon_B^e(F))$$

- Dilation of the erosion
- Removes small objects

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

(basic opening by 3×3 square SE: original image)

Opening

$$\gamma_B^e(F) = \delta_B^e(\varepsilon_B^e(F))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

- Dilation of the erosion
- Removes small objects

(basic opening by 3×3 square SE: eroded)

Opening

$$\gamma_B^e(F) = \delta_B^e(\varepsilon_B^e(F))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

- Dilation of the erosion
- Removes small objects

(basic opening by 3×3 square SE: eroded and dilated)

Closing

$$\phi_B^e(F) = \varepsilon_B^e(\delta_B^e(F))$$

- Erosion of the dilation

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

Closing

$$\phi_B^e(F) = \varepsilon_B^e(\delta_B^e(F))$$

- Erosion of the dilation
- Removes small holes

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

Closing

$$\phi_B^e(F) = \varepsilon_B^e(\delta_B^e(F))$$

- Erosion of the dilation
- Removes small holes

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

(basic closing by 3×3 square SE: original image)

Closing

$$\phi_B^e(F) = \varepsilon_B^e(\delta_B^e(F))$$

- Erosion of the dilation
- Removes small holes

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

(basic closing by 3×3 square SE: dilated)

Closing

$$\phi_B^e(F) = \varepsilon_B^e(\delta_B^e(F))$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

- Erosion of the dilation
- Removes small holes

(basic closing by 3×3 square SE: dilated and eroded)

Top-hat

$$\tau_B^e(F) = F - \gamma_B^e(F)$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

- Removes a particular feature from the image

Top-hat

$$\tau_B^e(F) = F - \gamma_B^e(F)$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

- Removes a particular feature from the image
- Example: edge detection using top-hat filter

(edge detection by top-hat: original image)

Top-hat

$$\tau_B^e(F) = F - \gamma_B^e(F)$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

- Removes a particular feature from the image
- Example: edge detection using top-hat filter

(edge detection by top-hat: erosion by 3×3 square)

Top-hat

$$\tau_B^e(F) = F - \gamma_B^e(F)$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

- Removes a particular feature from the image
- Example: edge detection using top-hat filter

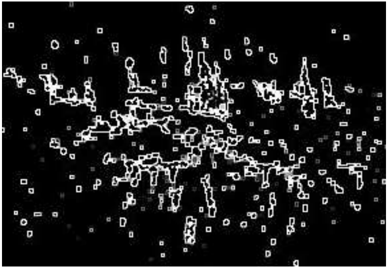
(edge detection by top-hat: opening by 3×3 square)

Top-hat

$$\tau_B^e(F) = F - \gamma_B^e(F)$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

- Removes a particular feature from the image
- Example: edge detection using top-hat filter



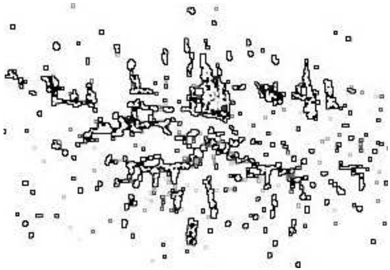
(edge detection by top-hat: top-hat with original image)

Top-hat

$$\tau_B^e(F) = F - \gamma_B^e(F)$$

B	Structuring element (SE)
e	SE dimension scaling factor (default: $e = 1$)
F	Image

- Removes a particular feature from the image
- Example: edge detection using top-hat filter



(edge detection by top-hat: inverted result)

Geodesic reconstruction

- One of the most common MM techniques

Geodesic reconstruction

- One of the most common MM techniques
- Instead of one image and a SE now two images are used

Geodesic reconstruction

- One of the most common MM techniques
- Instead of one image and a SE now two images are used
- *Marker* image is source image, *mask* image is max. or min. image (depending on operation)

Geodesic reconstruction

- One of the most common MM techniques
- Instead of one image and a SE now two images are used
- *Marker* image is source image, *mask* image is max. or min. image (depending on operation)
- Geodesic: Extracts connected components based on distance

Geodesic reconstruction

- One of the most common MM techniques
- Instead of one image and a SE now two images are used
- *Marker* image is source image, *mask* image is max. or min. image (depending on operation)
- Geodesic: Extracts connected components based on distance
- Can be used with different morphological operations

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F, G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}\left(\varepsilon_B(F)\right)\right)$$

B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion $\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F, G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}(\varepsilon_B(F))\right)$$

B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion $\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)

1. Erode marker image

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F,G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}\left(\varepsilon_B(F)\right)\right)$$

B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion $\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)

1. Erode marker image
2. Take maximum of eroded image and mask image

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F, G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}\left(\varepsilon_B(F)\right)\right)$$

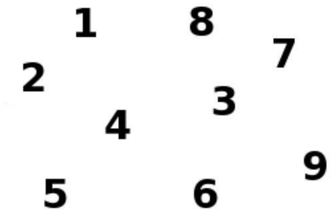
B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion
	$\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)

1. Erode marker image
2. Take maximum of eroded image and mask image
3. If image has been changed in this iteration goto 1

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F, G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}\left(\varepsilon_B(F)\right)\right)$$

B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion $\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)



(segment 1 and 4: original image)

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F,G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}\left(\varepsilon_B(F)\right)\right)$$

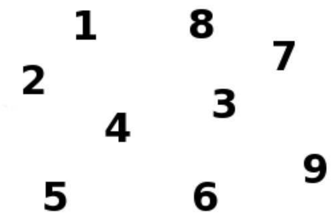
B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion $\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)

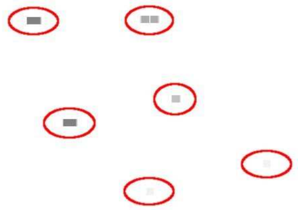
(segment 1 and 4: dilation by linear SE, length = 45 pixel, vertical)

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F,G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}\left(\varepsilon_B(F)\right)\right)$$

B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion $\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)



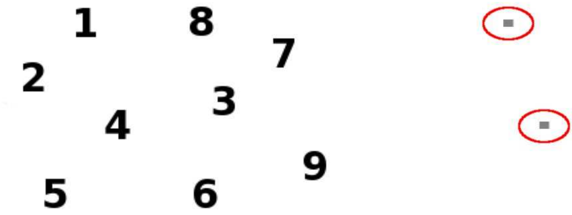


(segment 1 and 4: marked dilation result)

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F,G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}\left(\varepsilon_B(F)\right)\right)$$

B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion $\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)

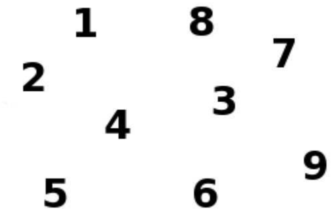


(segment 1 and 4: dilation by linear SE, length = 7, horizontal)

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F, G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}\left(\varepsilon_B(F)\right)\right)$$

B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion $\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)

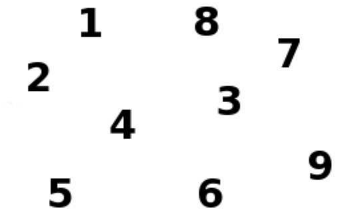


(segment 1 and 4: un-marked dilation result)

Geodesic reconstruction by erosion (geodesic closing)

$$\Phi(F,G) = \varepsilon_G^{(n)}(F) = \max\left(G, \varepsilon_G^{(n-1)}\left(\varepsilon_B(F)\right)\right)$$

B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
ε	Erosion $\varepsilon_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\varepsilon_{B,G}^{(n)}(F) = \varepsilon_{B,G}^{(n+1)}(F)$ holds)



(segment 1 and 4: geodesic reconstruction with original as mask)

Geodesic reconstruction by dilation (geodesic opening)

$$\Gamma(F, G) = \delta_G^{(n)}(F) = \min\left(G, \delta_G^{(n-1)}\left(\delta_B(F)\right)\right)$$

B	Isotropic structuring element
F	Image (Marker)
G	Image (Mask)
δ	Dilation
	$\delta_{B,G}^{(0)}(F) = F$
n	number of iterations until stability has been reached ($\delta_{B,G}^{(n)}(F) = \delta_{B,G}^{(n+1)}(F)$ holds)

1. Dilate marker image
2. Take minimum of dilated image and mask image
3. If image has been changed in this iteration goto 1

Structuring elements

- Based on observation of cracks specific SEs are chosen

Structuring elements

- Based on observation of cracks specific SEs are chosen
- Linear SE

Structuring elements

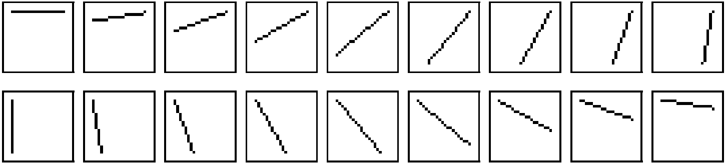
- Based on observation of cracks specific SEs are chosen
- Linear SE
- SE length: 12 pixel

Structuring elements

- Based on observation of cracks specific SEs are chosen
- Linear SE
- SE length: 12 pixel
- Degree of rotation: every 10° from 0° to 180°

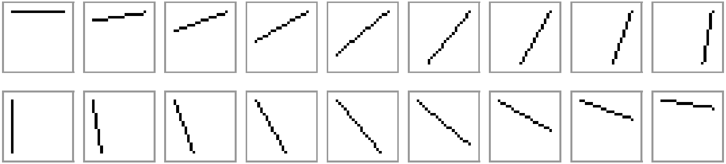
Structuring elements

- Based on observation of cracks specific SEs are chosen
- Linear SE
- SE length: 12 pixel
- Degree of rotation: every 10° from 0° to 180°



Structuring elements

- Based on observation of cracks specific SEs are chosen
- Linear SE
- SE length: 12 pixel
- Degree of rotation: every 10° from 0° to 180°



- Filters have been chosen for dark features

Linear filters

- Pictures of zebras and dalmatians both have black and white pixels

Linear filters

- Pictures of zebras and dalmatians both have black and white pixels
- They appear in about the same amount

Linear filters

- Pictures of zebras and dalmatians both have black and white pixels
- They appear in about the same amount
- Difference in order and characteristic appearance of groups

Linear filters

- Pictures of zebras and dalmatians both have black and white pixels
- They appear in about the same amount
- Difference in order and characteristic appearance of groups
- Linear filters are means to detect these specific characteristics

Linear filters

- Pictures of zebras and dalmatians both have black and white pixels
- They appear in about the same amount
- Difference in order and characteristic appearance of groups
- Linear filters are means to detect these specific characteristics
- Each pixel is set to a weighted sum of its and its neighbours' values (convolution)

Linear filters

- Pictures of zebras and dalmatians both have black and white pixels
- They appear in about the same amount
- Difference in order and characteristic appearance of groups
- Linear filters are means to detect these specific characteristics
- Each pixel is set to a weighted sum of its and its neighbours' values (convolution)
- Weights defined as matrix (kernel)

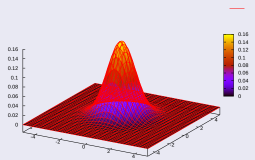
Linear filters

- Pictures of zebras and dalmatians both have black and white pixels
- They appear in about the same amount
- Difference in order and characteristic appearance of groups
- Linear filters are means to detect these specific characteristics
- Each pixel is set to a weighted sum of its and its neighbours' values (convolution)
- Weights defined as matrix (kernel)
- Here: edge detection

Gaussian

- Smoothing an image

Gaussian kernel

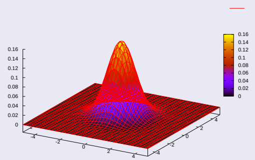


$$\mathcal{G}_\sigma(x, y) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{(x^2+y^2)}{2\sigma^2}\right)$$

Gaussian

- Smoothing an image
- Discrete Gaussian kernel from Gaussian function

Gaussian kernel



$$\mathcal{G}_\sigma(x, y) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{(x^2+y^2)}{2\sigma^2}\right)$$

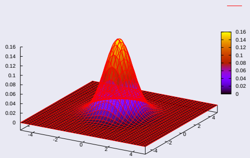
$$G_1 = \begin{bmatrix} \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \\ \frac{2}{16} & \frac{4}{16} & \frac{2}{16} \\ \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \end{bmatrix}$$

Gaussian

- Smoothing an image
- Discrete Gaussian kernel from Gaussian function

(a) Original (b) Gaussian

Gaussian kernel



$$\mathcal{G}_\sigma(x,y) = \frac{1}{2\pi\sigma^2}\, exp\left(-\frac{(x^2+y^2)}{2\sigma^2}\right)$$

$$G_1 = \begin{bmatrix} \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \\ \frac{2}{16} & \frac{4}{16} & \frac{2}{16} \\ \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \end{bmatrix}$$

Laplacian of Gaussian

- Classic method for edge detection

Laplacian of Gaussian kernel

Laplacian of Gaussian

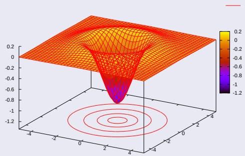
- Classic method for edge detection
- Laplacian operator:
  $(\nabla^2 f)(x,y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$

Laplacian of Gaussian kernel

Laplacian of Gaussian

- Classic method for edge detection
- Laplacian operator: $(\nabla^2 f)(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$
- Natural to smooth before applying laplacian $\Rightarrow$ Gaussian as function

Laplacian of Gaussian kernel

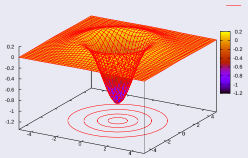


$$\mathcal{L}_\sigma = \frac{(x^2+y^2-2\sigma^2)}{\sigma^4} exp\left(-\frac{(x^2+y^2)}{(2\sigma^2)}\right)$$

Laplacian of Gaussian

- Classic method for edge detection
- Laplacian operator: $(\nabla^2 f)(x,y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$
- Natural to smooth before applying laplacian $\Rightarrow$ Gaussian as function
- $LoG_\sigma^w(F) = F \circ L_\sigma^w$ (F convolved with L)

Laplacian of Gaussian kernel

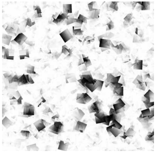


$$\mathcal{L}_\sigma = \frac{(x^2+y^2-2\sigma^2)}{\sigma^4} exp\left(-\frac{(x^2+y^2)}{(2\sigma^2)}\right)$$

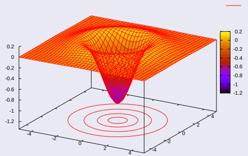
$$L_1^5 = \begin{bmatrix} -1 & -3 & -3 & -3 & -1 \\ -3 & 0 & 6 & 0 & -3 \\ -3 & 6 & 21 & 6 & -3 \\ -3 & 0 & 6 & 0 & -3 \\ -1 & -3 & -3 & -3 & -1 \end{bmatrix}$$

Laplacian of Gaussian

- Classic method for edge detection
- Laplacian operator: $(\nabla^2 f)(x,y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$
- Natural to smooth before applying laplacian $\Rightarrow$ Gaussian as function
- $LoG_\sigma^w(F) = F \circ L_\sigma^w$ (F convolved with L)



Laplacian of Gaussian kernel



$$\mathcal{L}_\sigma = \frac{(x^2+y^2-2\sigma^2)}{\sigma^4} exp\left(-\frac{(x^2+y^2)}{(2\sigma^2)}\right)$$

$$L_1^5 = \begin{bmatrix} -1 & -3 & -3 & -3 & -1 \\ -3 & 0 & 6 & 0 & -3 \\ -3 & 6 & 21 & 6 & -3 \\ -3 & 0 & 6 & 0 & -3 \\ -1 & -3 & -3 & -3 & -1 \end{bmatrix}$$

Processing pipeline structure

- Usage of mathematical morphology (MM) and linear filters (LF)
- Results in binary crack map
- Basic 3-step processing pipeline

1. Preprocessing (contrast enhancement)
2. Enhancement of cracks (MM and LF)
3. Segmentation of cracks (MM alternating filters)

Preprocessing

- Goal: Enhance contrast between cracks and background

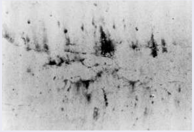
Preprocessing pipeline

Preprocessing

- Goal: Enhance contrast between cracks and background

0 Original grayscale image

Preprocessing pipeline

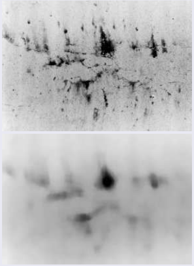


Preprocessing

- Goal: Enhance contrast between cracks and background

0. Original grayscale image
1. Median (15×15)

Preprocessing pipeline

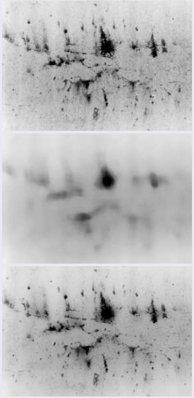


Preprocessing

- Goal: Enhance contrast between cracks and background

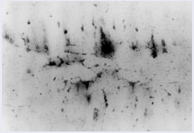
0. Original grayscale image
1. Median (15×15)
2. Compare foreground (original) and background (median) image, take minimum

Preprocessing pipeline



Crack enhancement

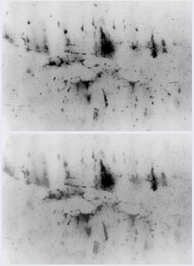
Enhancement pipeline



Crack enhancement

1. Closing by Reconstruction
   $F_{Cl} = \Phi\left(\min_{i=1,\ldots,18}\{\phi_{B_i}(F_0)\}\right)$

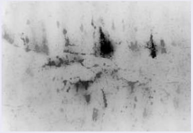
Enhancement pipeline



Crack enhancement

1. Closing by Reconstruction
   $F_{Cl} = \Phi\left(\min_{i=1,\ldots,18}\{\phi_{B_i}(F_0)\}\right)$

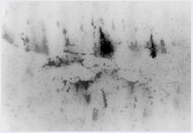
Enhancement pipeline



Crack enhancement

1. Closing by Reconstruction
   $F_{Cl} = \Phi\left(\min_{i=1,\ldots,18}\{\phi_{B_i}(F_0)\}\right)$
2. Sum of top-hats
   $$F_{th} = \sum_{i=0}^{17} \tau_{B_i}(F_{Cl}) = \sum_{i=0}^{17} (F_{Cl} - \gamma_{B_i}(F))$$
   Wrong formula (white objects)!

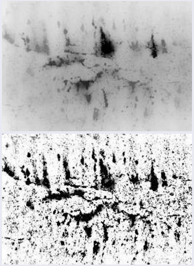
Enhancement pipeline



Crack enhancement

1. Closing by Reconstruction
   $F_{Cl} = \Phi\left(\min_{i=1,\ldots,18}\{\phi_{B_i}(F_0)\}\right)$
2. Sum of top-hats
   $$F_{th} = \left(\sum_{i=0}^{17} (\phi_{B_i}(F) - F_{Cl})\right)^{-1}$$
   Very noisy results, omitted.

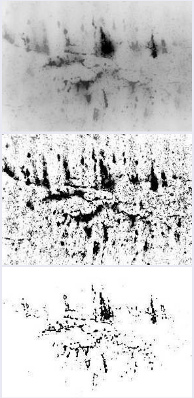
Enhancement pipeline



Crack enhancement

1. Closing by Reconstruction
   $F_{Cl} = \Phi\left(\min_{i=1,\dots,18}\{\phi_{B_i}(F_0)\}\right)$
2. Sum of top-hats
   $$F_{th} = \left(\sum_{i=0}^{17}\left(\phi_{B_i}(F) - F_{Cl}\right)\right)^{-1}$$
   Very noisy results, omitted.
3. Laplacian of Gaussian $F_{lap} = LoG_2^{12}(F_{Cl})$

Enhancement pipeline



Crack detection and segmentation

- Final segmentation of cracks

Segmentation pipeline

Crack detection and segmentation

- Final segmentation of cracks
- Alternating MM filters

Segmentation pipeline

Crack detection and segmentation

- Final segmentation of cracks
- Alternating MM filters

1. Closing by Reconstruction
   $F_1 = \Phi\left(\min_{i=1,\dots,18}\{\phi_{B_i}(F_{lap})\}\right)$

Segmentation pipeline

Crack detection and segmentation

- Final segmentation of cracks
- Alternating MM filters

1. Closing by Reconstruction
   $F_1 = \Phi\left(\min_{i=1,\ldots,18}\{\phi_{B_i}(F_{lap})\}\right)$

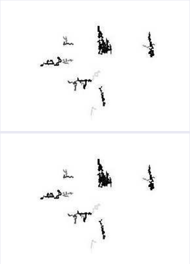
Segmentation pipeline

Crack detection and segmentation

- Final segmentation of cracks
- Alternating MM filters

1. Closing by Reconstruction
   $F_1 = \Phi\left(\min_{i=1,\dots,18}\{\phi_{B_i}(F_{lap})\}\right)$
2. Opening by Reconstruction
   $F_2 = \Gamma\left(\max_{i=1,\dots,18}\{\gamma_{B_i}(F_1)\}\right)$

Segmentation pipeline

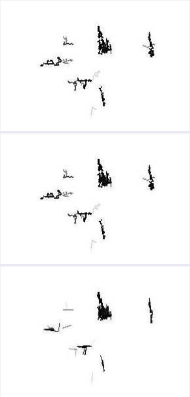


Crack detection and segmentation

- Final segmentation of cracks
- Alternating MM filters

1. Closing by Reconstruction
   $F_1 = \Phi\left(\min_{i=1,\ldots,18}\{\phi_{B_i}(F_{lap})\}\right)$
2. Opening by Reconstruction
   $F_2 = \Gamma\left(\max_{i=1,\ldots,18}\{\gamma_{B_i}(F_1)\}\right)$
3. Large closing with double scale
   $F_{final} = \left(\min_{i=1,\ldots,18}\{\phi^2_{B_i}(F_2)\}\right)$

Segmentation pipeline



Criteria for parameter selection

- Parameters: S length of SE in pixels, D degree of rotations

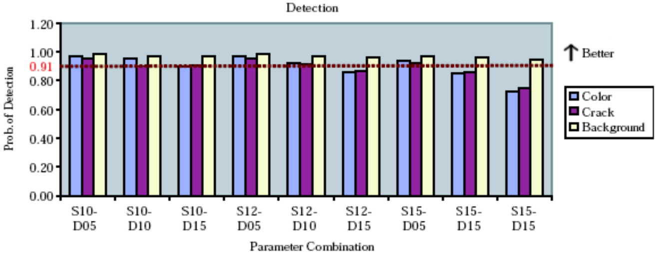
Criteria for parameter selection

- Parameters: S length of SE in pixels, D degree of rotations
- Goal: false positive rate below 7%, false negative rate below 2%

Criteria for parameter selection

- Parameters: S length of SE in pixels, D degree of rotations
- Goal: false positive rate below 7%, false negative rate below 2%

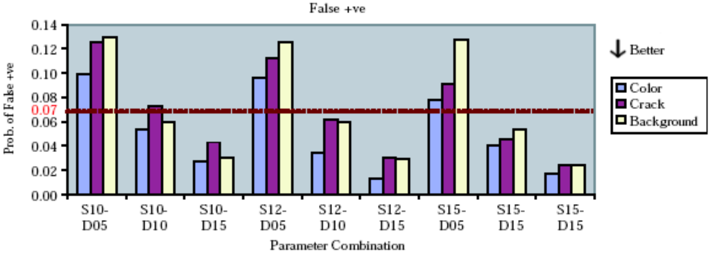
- Probability of detection



Criteria for parameter selection

- Parameters: S length of SE in pixels, D degree of rotations
- Goal: false positive rate below 7%, false negative rate below 2%

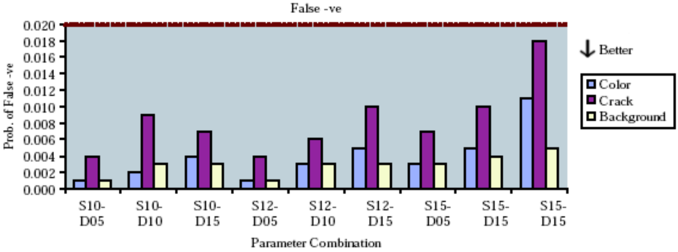
- Probability of false positive (crack detected where is none)



Criteria for parameter selection

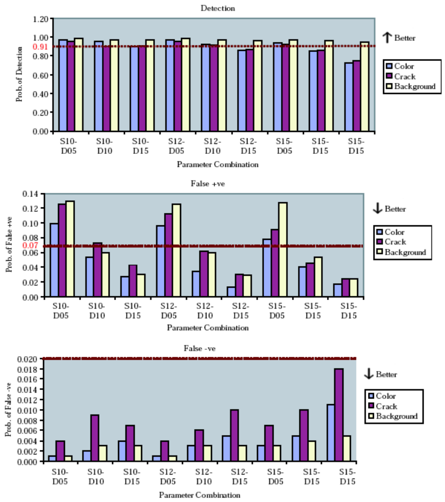
- Parameters: S length of SE in pixels, D degree of rotations
- Goal: false positive rate below 7%, false negative rate below 2%

- Probability of false negative (crack not detected)



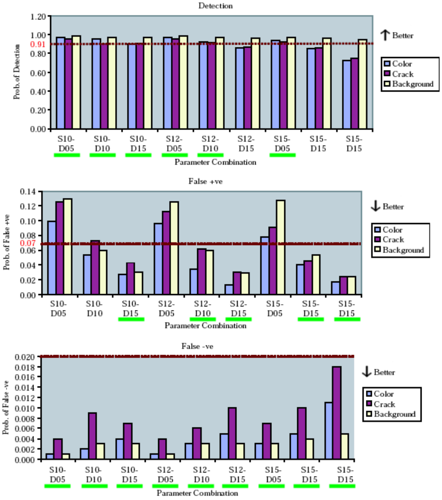
Criteria for parameter selection

- Probability of detection
- Probability of false positive
- Probability of false negative



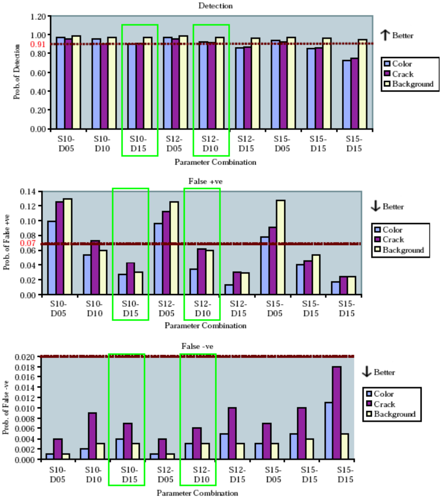
Criteria for parameter selection

- Probability of detection
- Probability of false positive
- Probability of false negative



Criteria for parameter selection

- Probability of detection
- Probability of false positive
- Probability of false negative
- Best parameters in paper: SE length $S = 12$ pixel and a degree of rotations $D =$ every 10°



Criteria for comparison to different approaches

- Comparison based on individual evaluation of approaches

Criteria for comparison to different approaches

- Comparison based on individual evaluation of approaches
- Ground truth by manually segmenting test images (reference)

Criteria for comparison to different approaches

- Comparison based on individual evaluation of approaches
- Ground truth by manually segmenting test images (reference)
- Completeness $\approx \frac{\text{\# matched crack pixels of ref.}}{\text{\# crack pixels of reference}}$ (optimal: 1)

Criteria for comparison to different approaches

- Comparison based on individual evaluation of approaches
- Ground truth by manually segmenting test images (reference)
- Completeness $\approx \frac{\text{\# matched crack pixels of ref.}}{\text{\# crack pixels of reference}}$ (optimal: 1)
- Correctness $\approx \frac{\text{\# matched crack pixels of extraction}}{\text{\# crack pixels of extraction}}$ (optimal: 1)

Criteria for comparison to different approaches

- Comparison based on individual evaluation of approaches
- Ground truth by manually segmenting test images (reference)
- Completeness $\approx \frac{\text{\# matched crack pixels of ref.}}{\text{\# crack pixels of reference}}$ (optimal: 1)
- Correctness $\approx \frac{\text{\# matched crack pixels of extraction}}{\text{\# crack pixels of extraction}}$ (optimal: 1)
- Redundancy $\approx \frac{\text{\# matched crack pixels of extr.} - \text{\# matched pixels of ref.}}{\text{\# crack pixels of extraction}}$ (optimal: 0)

Criteria for comparison to different approaches

- Comparison based on individual evaluation of approaches
- Ground truth by manually segmenting test images (reference)
- Completeness $\approx \frac{\#\text{ matched crack pixels of ref.}}{\#\text{ crack pixels of reference}}$ (optimal: 1)
- Correctness $\approx \frac{\#\text{ matched crack pixels of extraction}}{\#\text{ crack pixels of extraction}}$ (optimal: 1)
- Redundancy $\approx \frac{\#\text{ matched crack pixels of extr.} - \#\text{ matched pixels of ref.}}{\#\text{ crack pixels of extraction}}$ (optimal: 0)
- Quality $\approx \frac{\text{compl}\cdot\text{corr}}{\text{compl}-\text{compl}\cdot\text{corr}+\text{corr}}$ (optimal: 1)

Criteria for comparison to different approaches

- Comparison based on individual evaluation of approaches
- Ground truth by manually segmenting test images (reference)
- Completeness $\approx \frac{\text{\# matched crack pixels of ref.}}{\text{\# crack pixels of reference}}$ (optimal: 1)
- Correctness $\approx \frac{\text{\# matched crack pixels of extraction}}{\text{\# crack pixels of extraction}}$ (optimal: 1)
- Redundancy $\approx \frac{\text{\# matched crack pixels of extr.} - \text{\# matched pixels of ref.}}{\text{\# crack pixels of extraction}}$ (optimal: 0)
- Quality $\approx \frac{\text{compl} \cdot \text{corr}}{\text{compl} - \text{compl} \cdot \text{corr} + \text{corr}}$ (optimal: 1)

- Parameters for other approaches not mentioned in paper

Different approaches

Otsu's thresholding

- Apply thresholds to detect cracks
- Separates a number of intensity classes
- Uses statistical methods to minimize variance in a class and at the same time maximize the variance between the classes

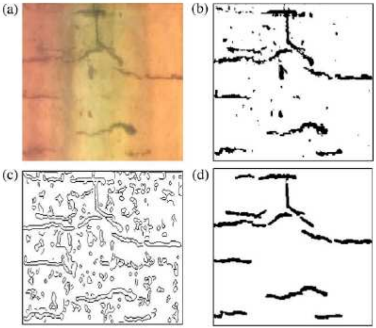


Fig. 13. Edge detection algorithms on crack pattern image: (a) original image, (b) Otsu's thresholding (c) Canny's edge detector, and (d) proposed approach.

Different approaches

Otsu's thresholding

- Apply thresholds to detect cracks
- Separates a number of intensity classes
- Uses statistical methods to minimize variance in a class and at the same time maximize the variance between the classes

Canny's edge detection

- Detect edges in the image between crack and background
- Uses linear filters (Gaussian and Sobel)
- Apply Gaussian, then apply a series of gradient filters to detect edges in different directions

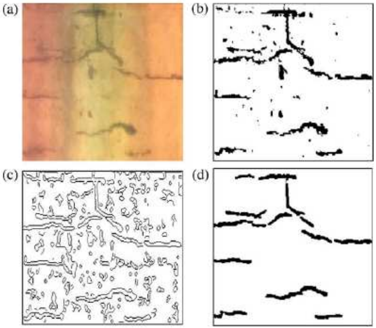


Fig. 13. Edge detection algorithms on crack pattern image: (a) original image, (b) Otsu's thresholding (c) Canny's edge detector, and (d) proposed approach.

Comparison to different approaches

- Otsu's thresholding
- Canny's edge detector
- No information about parameters in paper

Comparison to different approaches

- Otsu's thresholding
- Canny's edge detector
- No information about parameters in paper

Paper approach

Class	Cracks	Background	Color
Completeness	0.95	0.88	0.90
Correctness	0.98	0.94	0.91
Quality	0.93	0.83	0.83
Redundancy	0.00	-0.01	0.00

Comparison to different approaches

- Otsu's thresholding
- Canny's edge detector
- No information about parameters in paper

Paper approach

Class	Cracks	Background	Color
Completeness	0.95	0.88	0.90
Correctness	0.98	0.94	0.91
Quality	0.93	0.83	0.83
Redundancy	0.00	-0.01	0.00

Otsu's thresholding

Class	Cracks	Background	Color
Completeness	0.98	0.61	0.62
Correctness	0.37	0.45	0.08
Quality	0.37	0.35	0.08
Redundancy	0.22	0.23	0.24

Comparison to different approaches

- Otsu's thresholding
- Canny's edge detector
- No information about parameters in paper

Paper approach

Class	Cracks	Background	Color
Completeness	0.95	0.88	0.90
Correctness	0.98	0.94	0.91
Quality	0.93	0.83	0.83
Redundancy	0.00	-0.01	0.00

Otsu's thresholding

Class	Cracks	Background	Color
Completeness	0.98	0.61	0.62
Correctness	0.37	0.45	0.08
Quality	0.37	0.35	0.08
Redundancy	0.22	0.23	0.24

Canny's edge detector

Class	Cracks	Background	Color
Completeness	0.92	0.61	0.62
Correctness	0.20	0.44	0.07
Quality	0.20	0.34	0.07
Redundancy	0.15	0.17	0.14

Comparison to different approaches

- Otsu's thresholding
- Canny's edge detector
- No information about parameters in paper
- Very good evaluation results for proposed method in paper (not verified)

Paper approach

Class	Cracks	Background	Color
Completeness	0.95	0.88	0.90
Correctness	0.98	0.94	0.91
Quality	0.93	0.83	0.83
Redundancy	0.00	-0.01	0.00

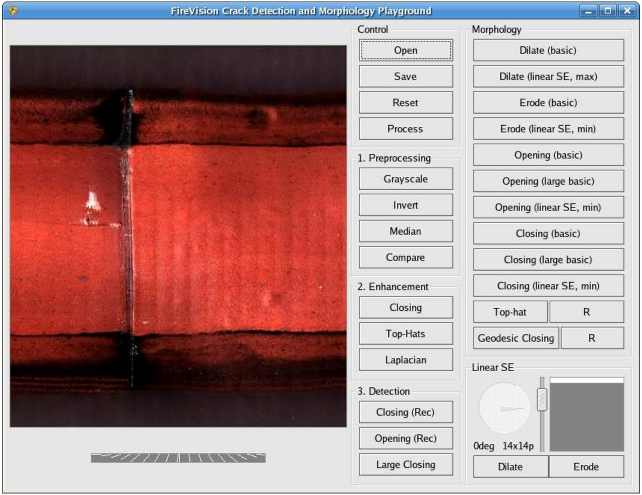
Otsu's thresholding

Class	Cracks	Background	Color
Completeness	0.98	0.61	0.62
Correctness	0.37	0.45	0.08
Quality	0.37	0.35	0.08
Redundancy	0.22	0.23	0.24

Canny's edge detector

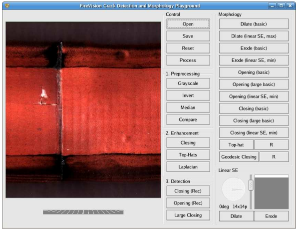
Class	Cracks	Background	Color
Completeness	0.92	0.61	0.62
Correctness	0.20	0.44	0.07
Quality	0.20	0.34	0.07
Redundancy	0.15	0.17	0.14

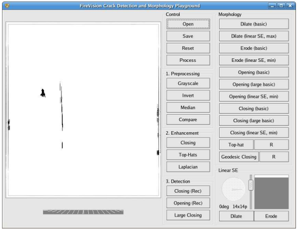
FireVision Crack Detection and Morphology Sandbox



FireVision Crack Detection and Morphology Sandbox

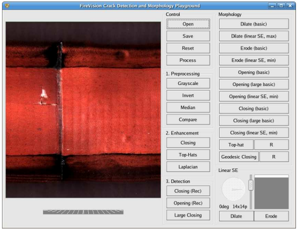
- Implemented in FireVision RoboCup vision framework from AllemaniACs RoboCup team

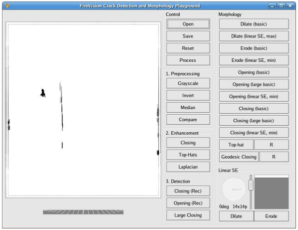




FireVision Crack Detection and Morphology Sandbox

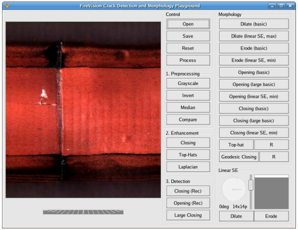
- Implemented in
  FireVision RoboCup vision framework
  from AllemaniACs RoboCup team
- Parameters adapted to sample images
  supplied by
  Institut für medizinische Informatik

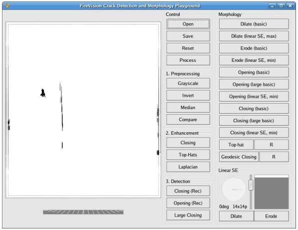




FireVision Crack Detection and Morphology Sandbox

- Implemented in
  FireVision RoboCup vision framework
  from AllemaniACs RoboCup team
- Parameters adapted to sample images
  supplied by
  Institut für medizinische Informatik
- Revealed several pieces of missing
  information and errors





Zana and Klein

Frederic Zana and Jean-Claude Klein: Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation. *IEEE Transactions on Image Processing*, 10(7):1010-1019, July 2001.

- Basically the template of the discussed paper

Zana and Klein

Frederic Zana and Jean-Claude Klein: Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation. *IEEE Transactions on Image Processing*, 10(7):1010-1019, July 2001.

- Basically the template of the discussed paper
- Proposed algorithm the same, just extracts brightest part of image

Zana and Klein

Frederic Zana and Jean-Claude Klein: Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation. *IEEE Transactions on Image Processing*, 10(7):1010-1019, July 2001.

- Basically the template of the discussed paper
- Proposed algorithm the same, just extracts brightest part of image
- More evaluation in discussed paper

Zana and Klein

Frederic Zana and Jean-Claude Klein: Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation. *IEEE Transactions on Image Processing*, 10(7):1010-1019, July 2001.

- Basically the template of the discussed paper
- Proposed algorithm the same, just extracts brightest part of image
- More evaluation in discussed paper
- Discussed paper *very* similar

Pros and Cons

- Some information not copied over from ZK paper

Pros and Cons

- Some information not copied over from ZK paper
- Wrong formulas (i.e. sum of top-hats)

Pros and Cons

- Some information not copied over from ZK paper
- Wrong formulas (i.e. sum of top-hats)
- Missing information: image sizes, typical crack length/width, parameters of other algorithms in evaluation, ...

Pros and Cons

- Some information not copied over from ZK paper
- Wrong formulas (i.e. sum of top-hats)
- Missing information: image sizes, typical crack length/width, parameters of other algorithms in evaluation, ...
- No quantitative data about typical human detection and error rates

Pros and Cons

- − Some information not copied over from ZK paper
- − Wrong formulas (i.e. sum of top-hats)
- − Missing information: image sizes, typical crack length/width, parameters of other algorithms in evaluation, ...
- − No quantitative data about typical human detection and error rates
- + Many pointers to interesting literature

Pros and Cons

- \- Some information not copied over from ZK paper
- \- Wrong formulas (i.e. sum of top-hats)
- \- Missing information: image sizes, typical crack length/width, parameters of other algorithms in evaluation, ...
- \- No quantitative data about typical human detection and error rates
- \+ Many pointers to interesting literature
- \+ Basics easy to reproduce

Pros and Cons

- Some information not copied over from ZK paper
- Wrong formulas (i.e. sum of top-hats)
- Missing information: image sizes, typical crack length/width, parameters of other algorithms in evaluation, ...
- No quantitative data about typical human detection and error rates
+ Many pointers to interesting literature
+ Basics easy to reproduce
+ Detailed evaluation section

1 Introduction

2 Morphology

3 Linear filters

4 Detection

5 Evaluation

6 Summary
- Conclusion
- End of Talk

Paper Resume

- Method to detect and segment cracks in underground pipeline images

Paper Resume

- Method to detect and segment cracks in underground pipeline images
- Presented approach uses mathematical morphology and curvature evaluation and makes use of a priori knowledge about crack structures

Paper Resume

- Method to detect and segment cracks in underground pipeline images
- Presented approach uses mathematical morphology and curvature evaluation and makes use of a priori knowledge about crack structures
- Evaluation has shown that the presented approach has good detection rates and low error rates (not verified)

Paper Resume

- Method to detect and segment cracks in underground pipeline images
- Presented approach uses mathematical morphology and curvature evaluation and makes use of a priori knowledge about crack structures
- Evaluation has shown that the presented approach has good detection rates and low error rates (not verified)
- Paper is derived from another paper and *very* similar

Questions?

Information compiled at
http://www.niemueller.de/uni/crackdet/