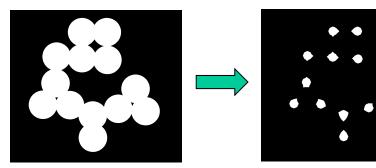
Morphological Image Processing

- Binary dilation and erosion
- Set-theoretic interpretation
- Opening, closing, morphological edge detectors
- Hit-miss filter
- Morphological filters for gray-level images
- Cascading dilations and erosions
- Rank filters, median filters, majority filters



NTEREST-POINT DETECTION

Feature extraction typically starts by finding the salient interest points in the image. For robust image matching, we desire interest points to be repeatable under perspective transformations (or, at least, scale changes, rotation, and translation) and real-world lighting variations. An example of feature extraction is illustrated in Figure 3. To achieve scale invariance, interest points are typically computed at multiple scales using an image pyramid [15]. To achieve rotation invariance, the patch around each interest point is canonically oriented in the direction of the dominant gradient. Illumination changes are compensated by normalizing the mean and standard deviation of the pixels of the gray values within each patch [16].

Binary image processing

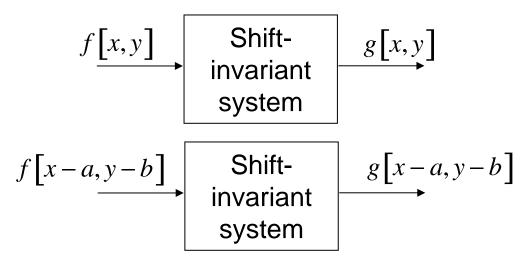
- Binary images are common
 - Intermediate abstraction in a gray-scale/color image analysis system
 - Thresholding/segmentation
 - Presence/absence of some image property
 - Text and line graphics, document image processing
- Representation of individual pixels as 0 or 1, convention:
 - foreground, object = 1 (white)
 - background = 0 (black)
- Processing by logical functions is fast and simple
- Shift-invariant logical operations on binary images: "morphological" image processing
- Morphological image processing has been generalized to gray-level images via level sets

Shift-invariance

• Assume that digital images f[x,y] and g[x,y] have infinite support

$$[x,y] \in \{\cdots,-2,-1,0,1,2,\cdots\} \times \{\cdots,-2,-1,0,1,2,\cdots\}$$

 \ldots then, for all integers a and b



Shift-invariance does <u>**not</u>** imply linearity (or vice versa).</u>

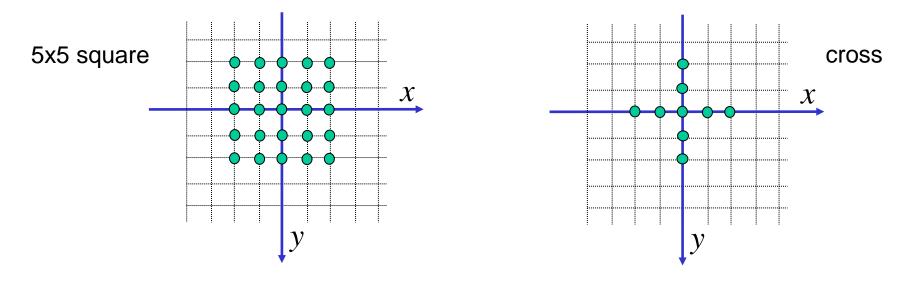
Structuring element

Neighborhood "window" operator

$$W\left\{f\left[x,y\right]\right\} = \left\{f\left[x-x',y-y'\right]:\left[x',y'\right]\in\prod_{xy}\right\}$$

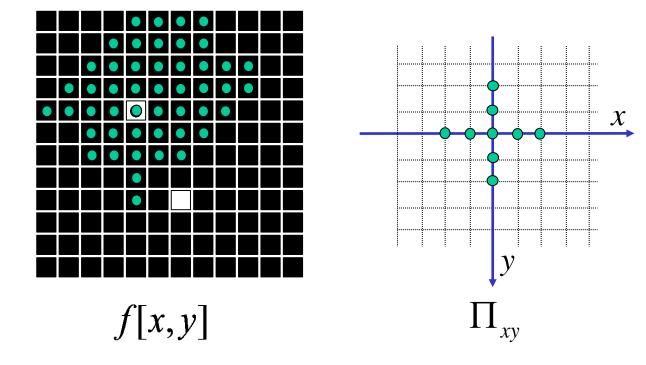
"structuring element"

• Example structuring elements Π_{xy} :

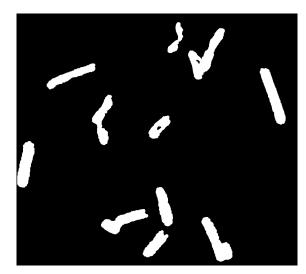


Binary dilation

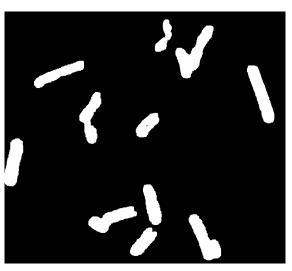
$$g[x,y] = OR[W\{f[x,y]\}] := dilate(f,W)$$



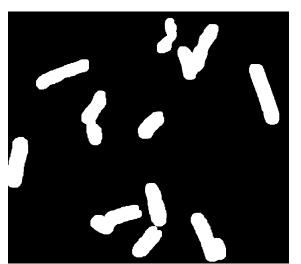
Binary dilation with square structuring element $g[x,y] = OR[W\{f[x,y]\}] := dilate(f,W)$



Original (701x781)



dilation with 3x3 structuring element



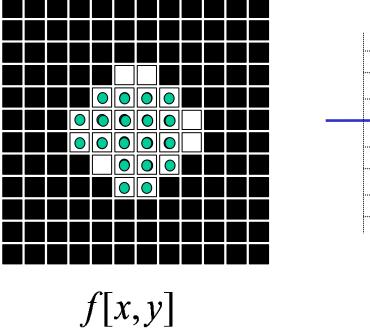
dilation with 7x7 structuring element

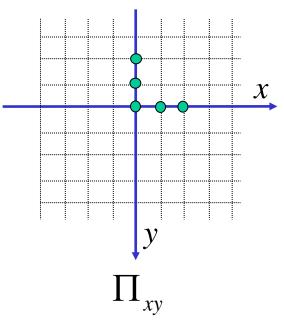
- Expands the size of 1-valued objects
- Smoothes object boundaries
- Closes holes and gaps

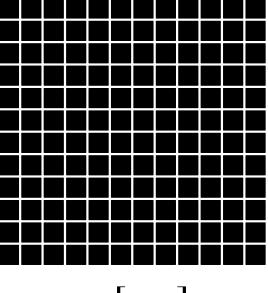


Binary erosion

$$g[x,y] = AND[W\{f[x,y]\}] \coloneqq erode(f,W)$$

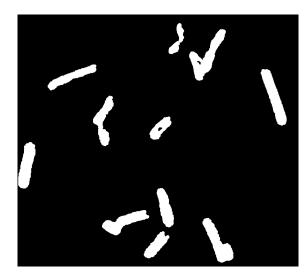




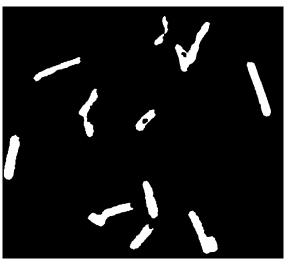


g[x,y]

Binary erosion with square structuring element $g[x,y] = AND[W\{f[x,y]\}] := erode(f,W)$



Original (701x781)



erosion with 3x3 structuring element



erosion with 7x7 structuring element

- Shrinks the size of 1-valued objects
- Smoothes object boundaries
- Removes peninsulas, fingers, and small objects



Relationship between dilation and erosion

Duality: erosion is dilation of the background

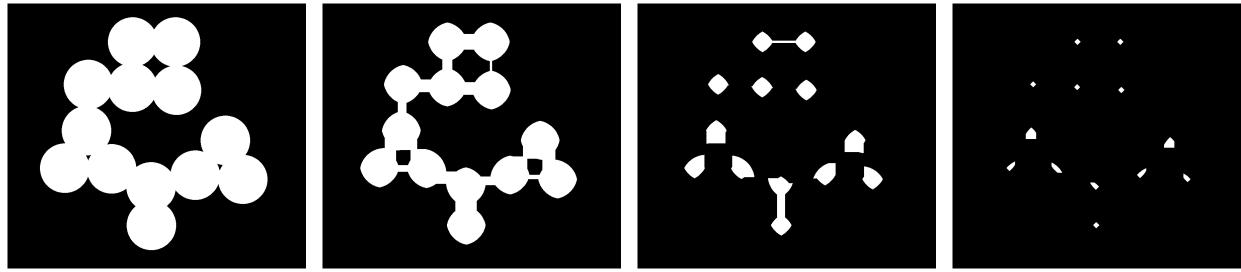
$$dilate(f,W) = NOT \left[erode(NOT \left[f \right], W) \right]$$
$$erode(f,W) = NOT \left[dilate(NOT \left[f \right], W) \right]$$

But: erosion is <u>not</u> the inverse of dilation

$$f[x,y] \neq erode(dilate(f,W),W)$$

$$\neq dilate(erode(f,W),W)$$

Example: blob separation/detection by erosion



Original binary image *Circles* (792x892)

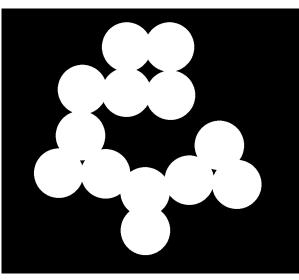
Erosion by 30x30 structuring element

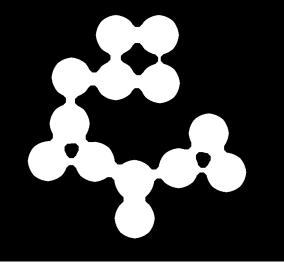
Erosion by 70x70 structuring element

Erosion by 96x96 structuring element



Example: blob separation/detection by erosion

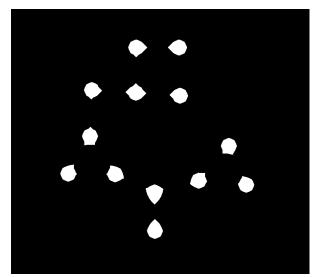




Original binary image *Circles* (792x892)

Erosion by disk-shaped structuring element Diameter=15

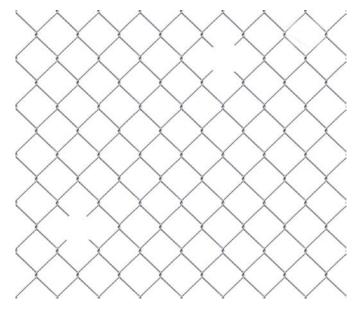
Erosion by disk-shaped structuring element Diameter=35



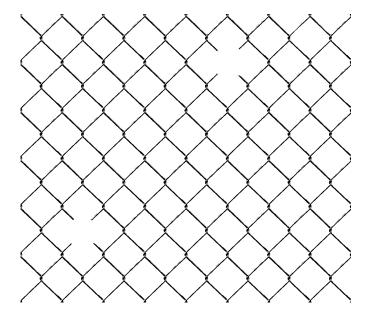
Erosion by disk-shaped structuring element Diameter=48



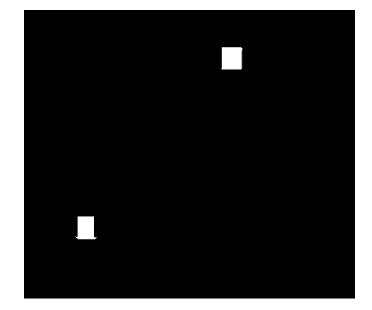
Example: chain link fence hole detection



Original grayscale image Fence (1023 x 1173)



Fence thresholded using Otsu's method



Erosion with 151x151 "cross" structuring element



Consider a binary image of infinite extent that is 0 everywhere, except for a single pixel f [3,3]=1. After performing dilation with an L-shaped structuring element (SE) of width 3 pixels, how does the resulting image look like?

(a) All zero, the single pixel has been removed.
(b) L-shaped SE shifted to position 3,3
(c) L-shaped SE rotated by 180 degrees and shifted to position 3,3
(d) L-shaped SE rotated by 180 degrees and shifted to position -3,-3
(e) None of the above

Answer the same question, but now performing erosion in lieu of dilation.

Set-theoretic interpretation

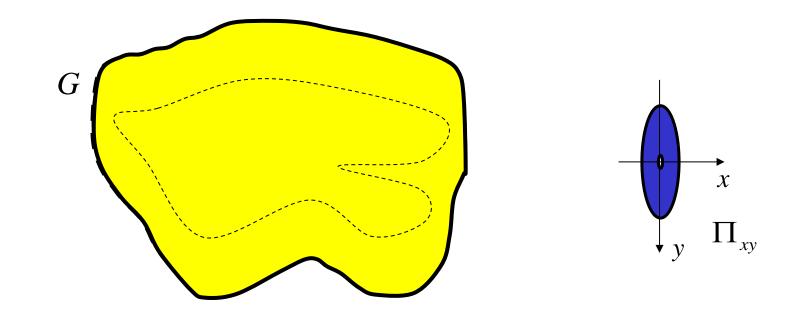
- Set of object pixels $F \equiv \left\{ (x, y): f(x, y) = 1 \right\}$ *Continuous* (x,y). *Works for discrete* [x,y] *in the same way.*
- Background: complement of foreground set $F^{c} \equiv \{(x, y) : f(x, y) = 0\}$
- Dilation is Minkowski set addition

$$G = F \oplus \Pi_{xy}$$

$$= \left\{ \left(x + p_x, y + p_y \right) : (x, y) \in F, \left(p_x, p_y \right) \in \Pi_{xy} \right\}$$

$$= \bigcup_{\left(p_x, p_y \right) \in \Pi_{xy}} F_{+\left(p_x, p_y \right)}$$
translation of F by vector $\left(p_x, p_y \right)$

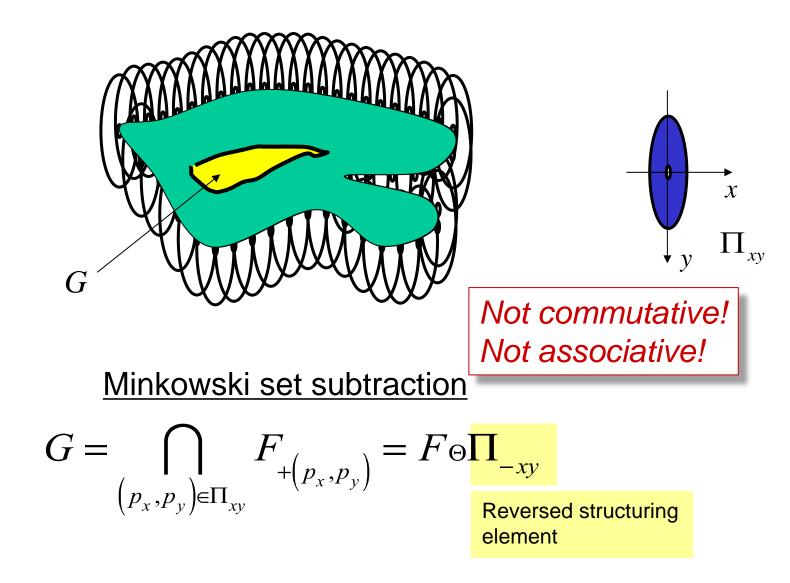
Set-theoretic interpretation: dilation



$$G = F \oplus \Pi_{xy}$$

= $\left\{ \left(x + p_x, y + p_y \right) : (x, y) \in F, \left(p_x, p_y \right) \in \Pi_{xy} \right\}$
= $\bigcup_{\left(p_x, p_y \right) \in \Pi_{xy}} F_{+\left(p_x, p_y \right)}$

Set-theoretic interpretation: erosion



Opening and closing

- Goal: smoothing without size change
- Open filter

Close filter

$$open(f,W) = dilate(erode(f,W),W)$$

$$close(f,W) = erode(dilate(f,W),W)$$

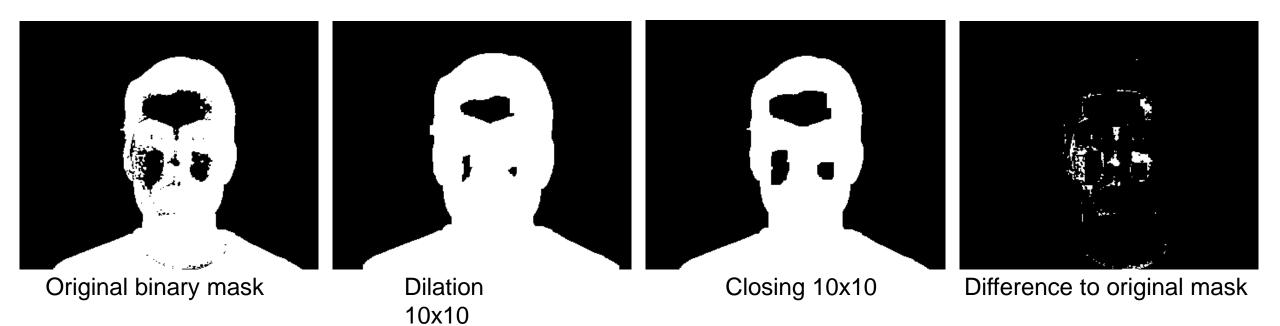
- Open filter and close filter are biased
 - Open filter removes small 1-regions
 - Close filter removes small 0-regions
 - Bias is often desired for enhancement or detection!
- Unbiased size-preserving smoothers

$$close - open(f, W) = close(open(f, W), W)$$

 $open - close(f, W) = open(close(f, W), W)$

• *close-open* and *open-close* are duals, but not inverses of each other.

Small hole removal by closing



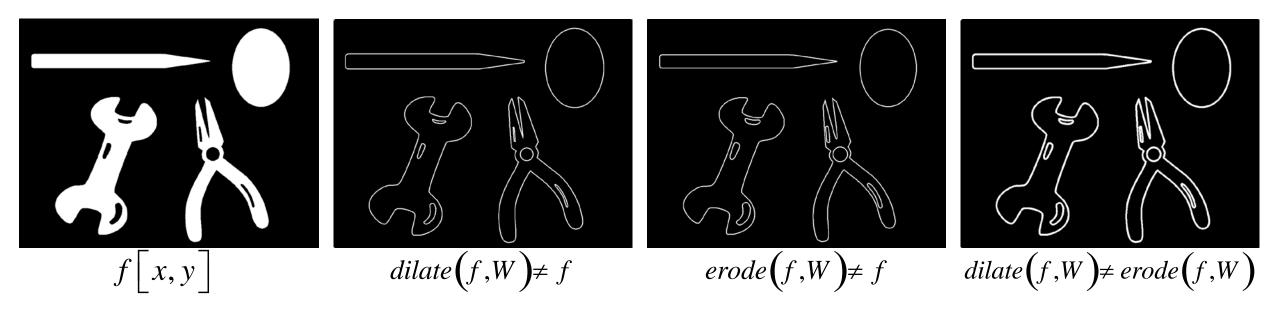


Which of the following holes in an object are filled by a morphological closing operation with a 10x10 square structuring element.

(a) 5x5 square

- (b) 6x6 square
- (c) 9x9 square
- (d) 10x10 square
- (e) 10x10 square rotated by 45 degrees

Morphological edge detectors





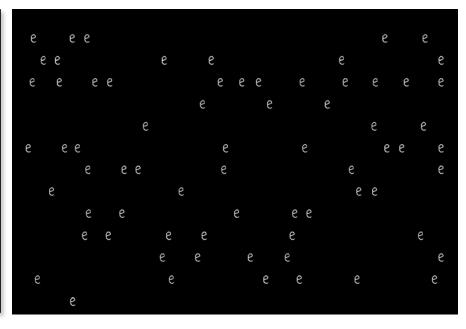
Binary image f

INTEREST-POINT DETECTION

1400

Feature extraction typically starts by finding the salient interest points in the image. For robust image matching, we desire interest points to be repeatable under perspective transformations (or, at least, scale changes, rotation, and translation) and real-world lighting variations. An example of feature extraction is illustrated in Figure 3. To achieve scale invariance, interest points are typically computed at multiple scales using an image pyramid [15]. To achieve rotation invariance, the patch around each interest point is canonically oriented in the direction of the dominant gradient. Illumination changes are compensated by normalizing the mean and standard deviation of the pixels of the gray values within each patch [16].

$$open(NOT[f],W) = dilate(erode(NOT[f],W),W)$$



2000

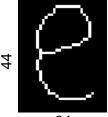


34

NTEREST-POINT DETECTION

Feature extraction typically starts by finding the salient interest points in the image. For robust image matching, we desire interest points to be repeatable under perspective transformations (or, at least, scale changes, rotation, and translation) and real-world lighting variations. An example of feature extraction is illustrated in Figure 3. To achieve scale invariance, interest points are typically computed at multiple scales using an image pyramid [15]. To achieve rotation invariance, the patch around each interest point is canonically oriented in the direction of the dominant gradient. Illumination changes are compensated by normalizing the mean and standard deviation of the pixels of the gray values within each patch [16].

Structuring element W ^a



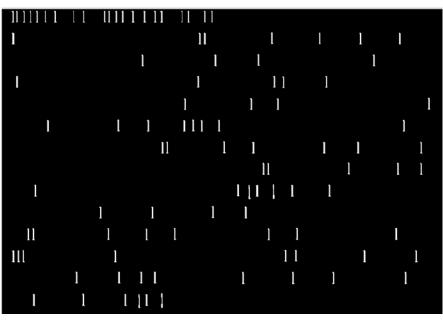
Binary image f

INTEREST-POINT DETECTION

1400

Feature extraction typically starts by finding the salient interest points in the image. For robust image matching, we desire interest points to be repeatable under perspective transformations (or, at least, scale changes, rotation, and translation) and real-world lighting variations. An example of feature extraction is illustrated in Figure 3. To achieve scale invariance, interest points are typically computed at multiple scales using an image pyramid [15]. To achieve rotation invariance, the patch around each interest point is canonically oriented in the direction of the dominant gradient. Illumination changes are compensated by normalizing the mean and standard deviation of the pixels of the gray values within each patch [16].

open(NOT[f],W) = dilate(erode(NOT[f],W),W)



2000



INTEREST-POINT DETECTION

Feature extraction typically starts by finding the salient interest points in the image. For robust image matching, we desire interest points to be repeatable under perspective transformations (or, at least, scale changes, rotation, and translation) and real-world lighting variations. An example of feature extraction is illustrated in Figure 3. To achieve scale invariance, interest points are typically computed at multiple scales using an image pyramid [15]. To achieve rotation invariance, the patch around each interest point is canonically oriented in the direction of the dominant gradient. Illumination changes are compensated by normalizing the mean and standard deviation of the pixels of the gray values within each patch [16].



18

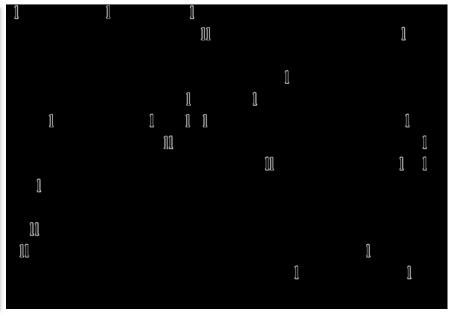
Hit-miss filter

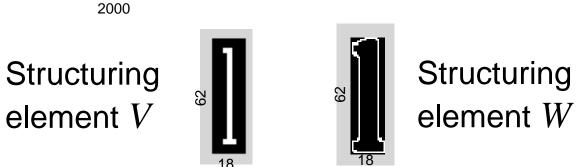
Binary image f

INTEREST-POINT DETECTION

1400

Feature extraction typically starts by finding the salient interest points in the image. For robust image matching, we desire interest points to be repeatable under perspective transformations (or, at least, scale changes, rotation, and translation) and real-world lighting variations. An example of feature extraction is illustrated in Figure 3. To achieve scale invariance, interest points are typically computed at multiple scales using an image pyramid [15]. To achieve rotation invariance, the patch around each interest point is canonically oriented in the direction of the dominant gradient. Illumination changes are compensated by normalizing the mean and standard deviation of the pixels of the gray values within each patch [16]. $dilate\left(erode(NOT[f],V) & erode(f,W),W\right)$



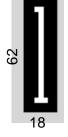


Hit-miss filter

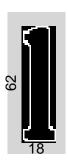
INTEREST-POINT DETECTION

Feature extraction typically starts by finding the salient interest points in the image. For robust image matching, we desire interest points to be repeatable under perspective transformations (or, at least, scale changes, rotation, and translation) and real-world lighting variations. An example of feature extraction is illustrated in Figure 3. To achieve scale invariance, interest points are typically computed at multiple scales using an image pyramid [15]. To achieve rotation invariance, the patch around each interest point is canonically oriented in the direction of the dominant gradient. Illumination changes are compensated by normalizing the mean and standard deviation of the pixels of the gray values within each patch [16].

Structuring element V



Structuring element W



Morphological filters for gray-level images

• Threshold sets of a gray-level image f[x,y]

$$T_{\theta} \Big(f \Big[x, y \Big] \Big) = \Big\{ \Big[x, y \Big] \colon f \Big[x, y \Big] \ge \theta \Big\}, \qquad -\infty < \theta < +\infty$$

Reconstruction of original image from threshold sets

$$f[x,y] = \sup \left\{ \theta : [x,y] \in \mathbf{T}_{\theta} (f[x,y]) \right\}$$

- Idea of morphological operators for multi-level (or continuous-amplitude) signals
 - Decompose into threshold sets
 - Apply binary morphological operator to each threshold set
 - Reconstruct via supremum operation
 - Gray-level operators thus obtained: *flat operators*
 - ➔ Flat morphological operators and thresholding are commutative

Dilation/erosion for gray-level images

- Explicit decomposition into threshold sets not required in practice
- Flat dilation operator: local maximum over window W

$$g[x,y] = \max\left\{W\left\{f[x,y]\right\}\right\} \coloneqq dilate(f,W)$$

Flat erosion operator: local minimum over window W

$$g[x,y] = \min\left\{W\left\{f[x,y]\right\}\right\} \coloneqq erode(f,W)$$

Binary dilation/erosion operators contained as special case

1-d illustration of erosion and dilation

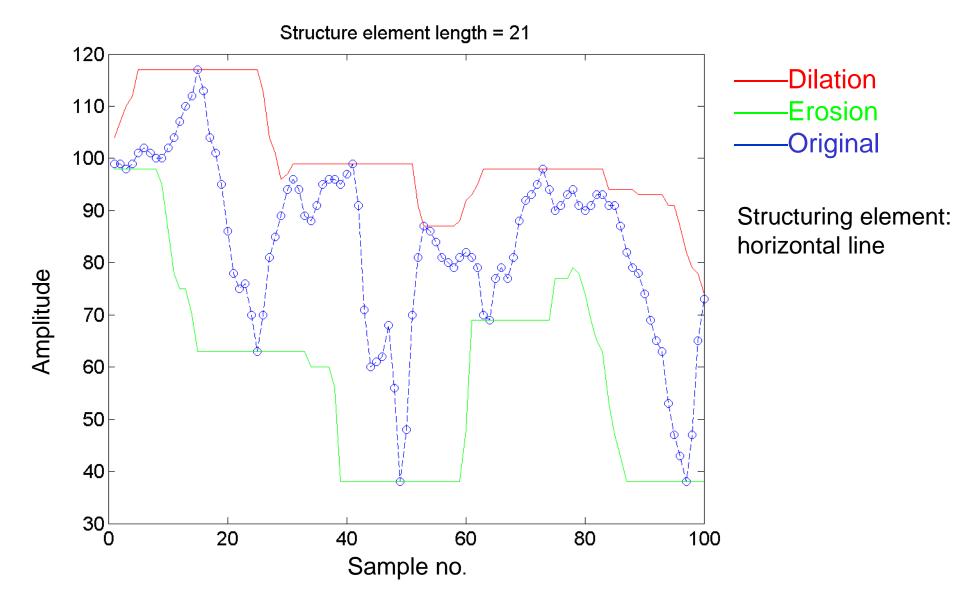


Image example



Original



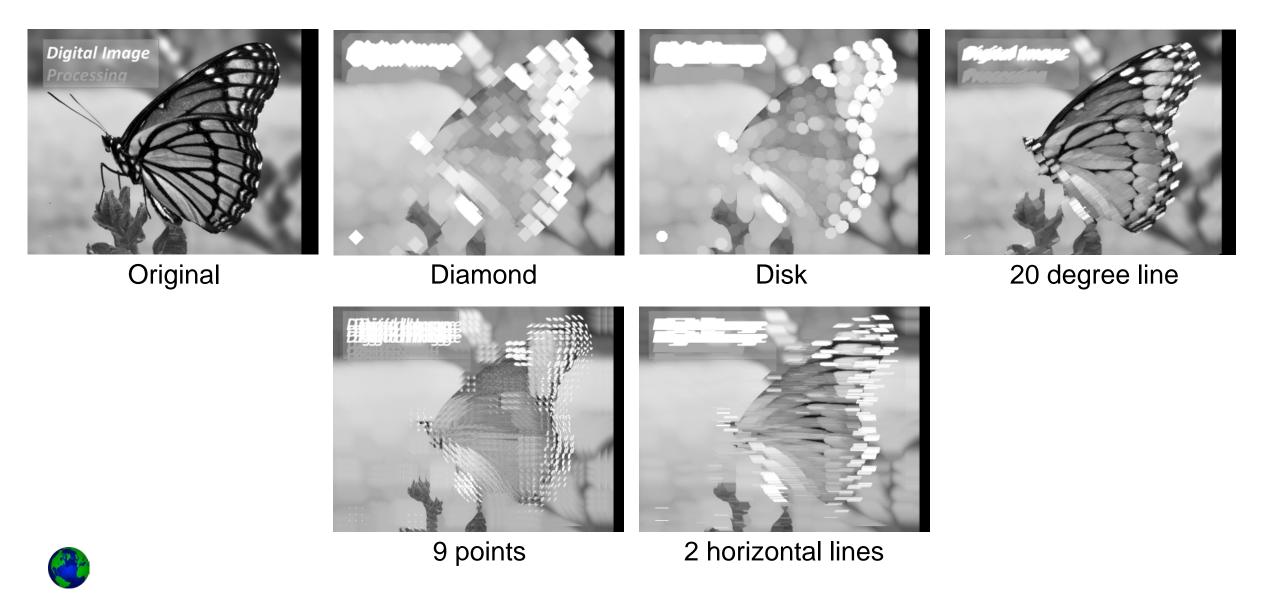
Dilation



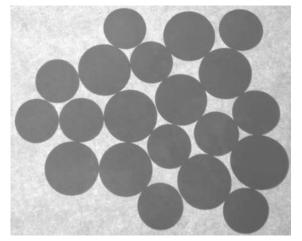




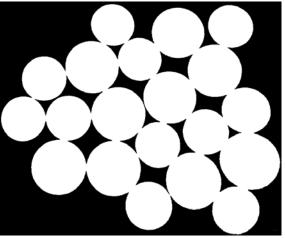
Flat dilation with different structuring elements



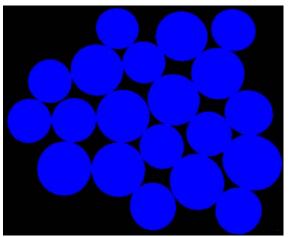
Example: counting coins



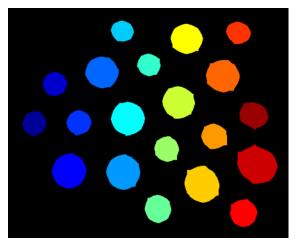
Original



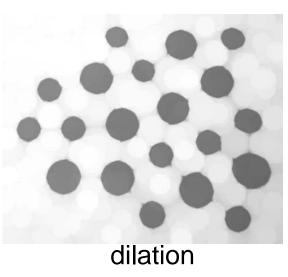
thresholded

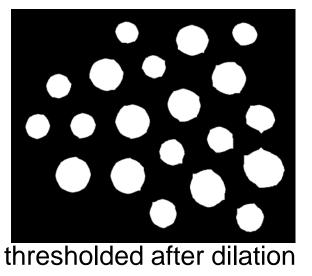


1 connected component



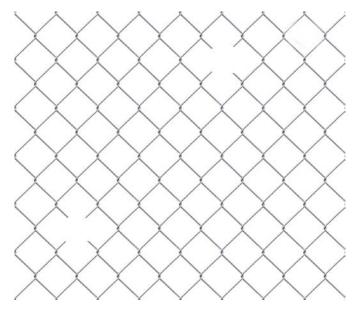
20 connected components



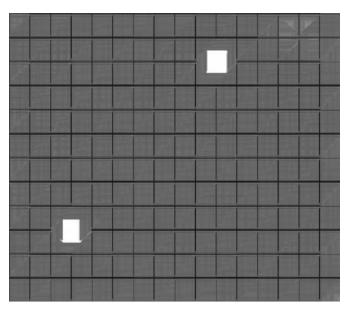




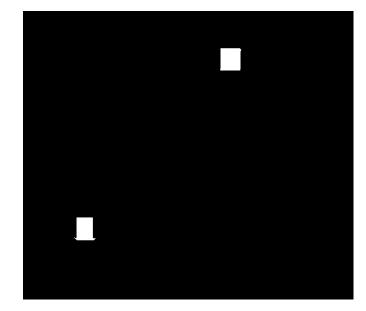
Example: chain link fence hole detection



Original grayscale image Fence (1023 x 1173)



Flat erosion with 151x151 "cross" structuring element



Binarized by Thresholding



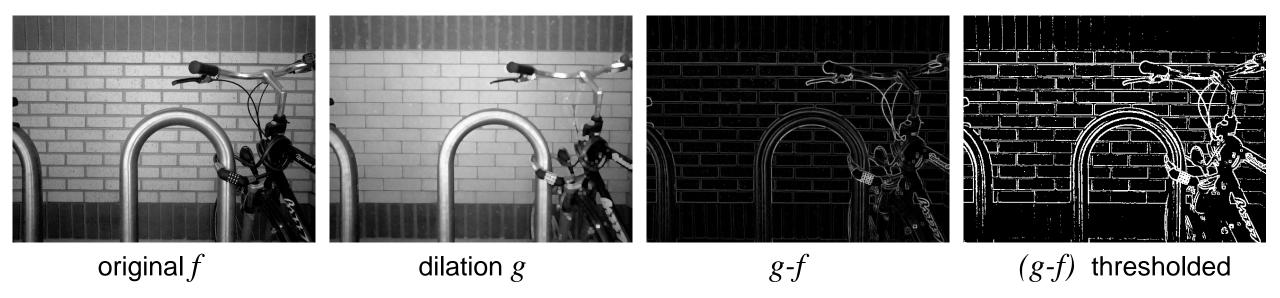
For the fence hole detection example, which method would you expect to give the better results in the presence of noise:

(a) Flat erosion with 151x151 cross, followed by thresholding

(b) Thresholding of the image, followed by erosion with a 151x151 cross

(c) Neither has an advantage over the other

Morphological edge detector





Beyond flat morphogical operators

General dilation operator

$$g[x,y] = \sup_{\alpha,\beta} \left\{ f[x-\alpha,y-\beta] + w[\alpha,\beta] \right\} = \sup_{\alpha,\beta} \left\{ w[x-\alpha,y-\beta] + f[\alpha,\beta] \right\}$$

Like linear convolution, with sup replacing summation, addition replacing multiplication

Dilation with "unit impulse"

$$d[\alpha,\beta] = \begin{cases} 0 & \alpha = \beta = 0 \\ -\infty & \text{else} \end{cases}$$

does not change input signal:

$$f[x,y] = \sup_{\alpha,\beta} \left\{ f[x-\alpha,y-\beta] + d[\alpha,\beta] \right\}$$

Flat dilation as a special case

• Find
$$w[\alpha,\beta]$$
 such that

$$f[x,y] = \sup_{\alpha,\beta} \left\{ f[x-\alpha,y-\beta] + w[\alpha,\beta] \right\} = dilate(f,W)$$

Answer:

$$w[\alpha,\beta] = \begin{cases} 0 & [\alpha,\beta] \in \Pi_{xy} \\ -\infty & \text{else} \end{cases}$$

Hence, write in general

$$g[x,y] = \sup_{\alpha,\beta} \left\{ f[x-\alpha, y-\beta] + w[\alpha,\beta] \right\}$$
$$= dilate(f,w) = dilate(w,f)$$

General erosion for gray-level images

General erosion operator

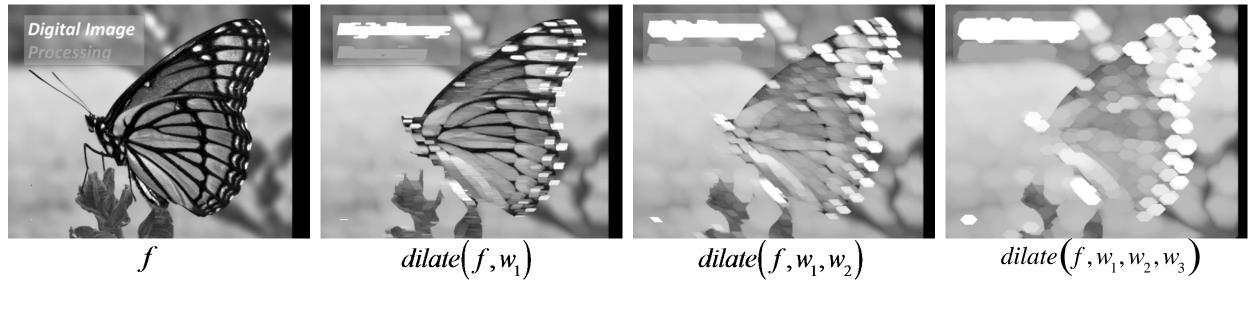
$$g[x,y] = \inf_{\alpha,\beta} \left\{ f[x-\alpha,y-\beta] - w[\alpha,\beta] \right\} = erode(f,w)$$

Dual of dilation

$$g[x,y] = \inf_{\alpha,\beta} \left\{ f[x-\alpha,y-\beta] - w[\alpha,\beta] \right\}$$
$$= -\sup_{\alpha,\beta} \left\{ -f[x-\alpha,y-\beta] + w[\alpha,\beta] \right\} = -dilate(-f,w)$$

Flat erosion contained as a special case

Cascaded dilations



$$dilate\left[dilate(f, w_1), w_2\right] = dilate(f, w)$$

where $w = dilate(w_1, w_2)$



Cascaded erosions

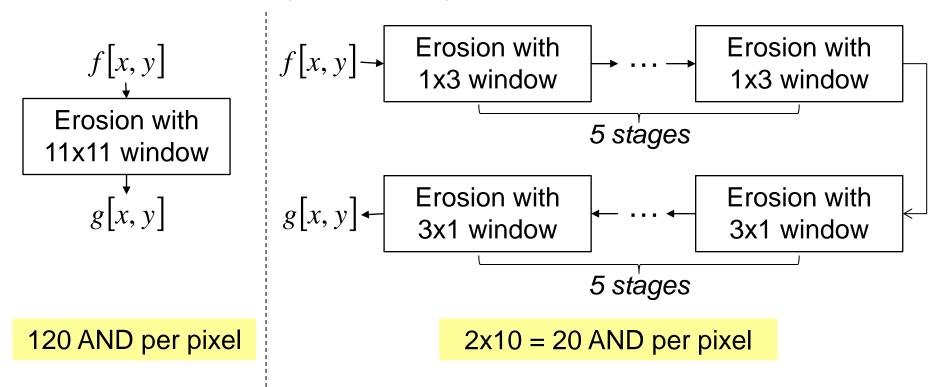
Cascaded erosions can be lumped into single erosion

$$erode \left[erode \left(f, w_1 \right), w_2 \right] = erode \left[-dilate \left(-f, w_1 \right), w_2 \right]$$
$$= -dilate \left[dilate \left(-f, w_1 \right), w_2 \right]$$
$$= -dilate \left(-f, w \right)$$
$$= erode \left(f, w \right)$$
where $w = dilate \left(w_1, w_2 \right)$

New structuring element (SE) is <u>not</u> the erosion of one SE by the other, but dilation.

Fast dilation and erosion

- Idea: build larger dilation and erosion operators by cascading simple, small operators
- Example: binary erosion by 11x11 window



Which of the following schemes performs dilation with a 5x5 square?

(a) Two cascaded dilations with a 3x3 structuring element

(b) Two cascaded dilations with a 1x3 structuring element, followed by two cascaded dilations with a 3x1 structuring element

(c) NOT operation applied to invert the image, followed by two cascaded erosions with a 1x3 structuring element, followed by two cascaded erosions with a 3x1 structuring element, followed by another NOT operation

Rank filters

- Generalisation of flat dilation/erosion: in lieu of min or max value in window, use the p-th ranked value
- Increases robustness against noise
- Best-known example: median filter for noise reduction
- Concept useful for both gray-level and binary images
- All rank filters are commutative with thresholding

Median filter

• Gray-level median filter

$$g[x,y] = median[W\{f[x,y]\}] \coloneqq median(f,W)$$

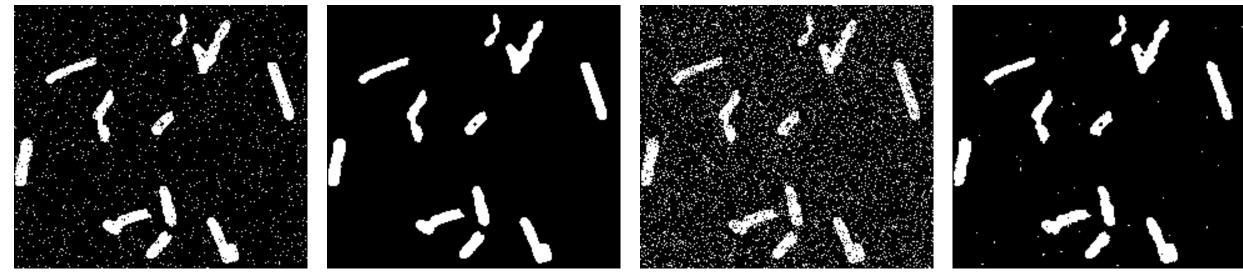
Binary images: majority filter

$$g[x,y] = MAJ[W\{f[x,y]\}] \coloneqq majority(f,W)$$

Self-duality

$$median(f,W) = -\left[median(-f,W)\right]$$
$$majority(f,W) = NOT\left[majority(NOT[f],W)\right]$$

Majority filter: example



Binary image with 5% 'Salt&Pepper' noise

3x3 majority filter

20% 'Salt&Pepper' noise

3x3 majority filter



Median filter: example



Original image

5% 'Salt&Pepper' noise

3x3 median filtering

7x7 median filtering



Example: non-uniform lighting compensation

GPD-ACCELERATED LOCAL TUNE-MAPPING FOR HIGH DYNAMIC RANGE IMAGES Octoor Dian', Jung Duan'', Guapting Olu"

thwestern University of Finance and Eco

ABSTRACT

I. INTRODUCTION

te is defined as the ratio invest luminance. The real world which suffer a limited dy, per color ch



same appropriately designed m mage [3] and [4] are p operators attempt to match the us, and match the

the displayed image and the se wes a technique based on a connodel, successfully simulating in effects like adaptation and color appearant presents an endod based on logarithmic compression of presents a method based on logarithmic compression of baninance values, initiating the human response to light Recently, [7] formulates the tone mapping problem as inization process and employs an adaptiv-ming strategy to obtain mapped images. Perechensive technique is still that of [8], which first res histogram equalization and then extends this idea opporate models of human contrast ser-

spatial acuity, and color sensitivity effects. Local tone mapping techniques use ing functions. [9-12] are based on the ng an image into layers ng them. Usually, layers with ssed to reduce the dy details are untouched or even en presents a method based on a mi nex theory of color vis





Original image 1632x1216 pixels

Dilation (local max) 61x61 structuring element

Rank filter 10st brightest pixel 61x61 structuring element



Example: non-uniform lighting compensation

a of Electrical Engineering, Stanford University ation Engineering, Southwestern University of Finance and Economics,

Fig.1. Sel



Tone mapping operators are usually classified as either

L INTRODUCTION

a very wide range of luminance (Fig. 1), ng 10 orders of magnitude. To reproduce devices, which suffer a limited dynamic s of magnitude. Radiance maps [1, 2], a sequence of low dynamic range same scene taken under different t. 1), are able to record the full ene in 32-bit floating-point number ere in 32-thir realing-point number reproduction devices such as CRT Local tone mapping techniques use spatially varying.

same appropriately designed mapping function to every pixel across the image. [3] and [4] are pioneering works The operators attempt to match the display brightness with real world sensations, and match the perceived contrast between the displayed image and the scene respectively. Later, [5] proposes a technique based on a comprehensive visual model, successfully simulating important visual a the lowest luminance. The real world are entry and ent presents a method based on logarithmic compression of luminance values, imitating the human response to light. is a challenge for conventional digital Recently, [7] formulates the tone mapping problem as a quantization process and employs an adaptive conscience learning strategy to obtain mapped images. Perhaps the most comprehensive technique is still that of [8], which first improves histogram equalization and then extends this idea to incorporate models of human contrast sensitivity, glare, spatial acuity, and color sensitivity effects

ally only 8-bit per color channel. mapping functions. [9-12] are based on the same principle any only sont per color channel. Reproduction is the process to of decomposing an image into layers and differently compressing than Usually the eproduction is the process to compressing them. Usually, layers with large features are strongly compressed to set while preserving as much of strongly compressed to reduce the dynamic range while layers of details are untouched or even enhanced to preserve e possible, E issue by presenting a novel details. [13] presents a method based on a multiscale version of the Refiney theory of rest. assue by presenting a novel apping method for displaying of the Retinex theory of color vision. [14] attempts to incorporate traditional photographic techniques to the digital the paper is as follows. In domain for reproducing HDR images. [15] compresses GPU. We describe our dynamic range through the manipulation of the gradient domain in the lonarithmic sector. GPU implementation in domain in the logarithmic space. More recently, [16] PC implementation in storage in the togenumber space. Note: recently, Linds experimental results proposes a novel method by adjusting the local histogram. There has been little published research to explore rendering HDR images on the GPU [17,18] and these

GPU-ACCELERATED FOCAL FONE-MAPPING FOR HIGH DYNAMIC RANGE IMAGES Ossian frant Juang Duan", Guoping Qiu

Department of Electrical Engineering, Stanford University "School of Feonomic Information Engineering, Southwestern University of Finance and Economics, Chengdu, China

"School of Computer Science. The University of Nottingham, UK

ABSTRACT

This paper presents a very fast local tone mapping method for displaying high dynamic range (HDR) images. Though (ksal tone mapping operators produce better local contrast details, they are usually slow. We have solved this oblem by designing a highly parallel algorithm, which can easily implemented on a Graphics Processing Unit (GPU) barvest high computational efficiency. At the same time,

roposed method mimics the local adaption mechanism human visual system and thus gives good results for a iety of images.

av Termo- Local tone mapping, high dynamic rallel computation, GPU, CUDA

I. INTRODUCTION

ee of a scene or an image is defined as the ratio st to the lowest luminance. The real world tave a very wide range of luminance (Fig. 1). eeding 10 orders of magnitude. To reproduce sents a challenge for conventional digital av devices, which suffer a limited dynamic rders of magnitude. Radiance maps [1, 2], ing a sequence of low dynamic range the same scene taken under different (Fig. 1), are able to record the full scene in 32-bit floating-point number reproduction devices such as CRT usually only 8-bit per color channel. Reproduction is the process to nge of the radiance maps to fit into es, while preserving as much of as possible.

his issue by presenting a novel mapping method for displaying of the paper is as follows. In view previous works of tone the GPU. We describe our GPU implementation in sents experimental results



2. REVIEW OF TONE MAPPING METHODS

Tone mapping operators are usually classified as either global or local. Global tone mapping techniques apply the same appropriately designed mapping function to every pixel across the image. [3] and [4] are pioneering works. The operators attempt to match the display brightness with real world sensations, and match the perceived contrast between the displayed image and the scene respectively. Later, [5] proposes a technique based on a comprehensive visual model, successfully simulating important visual effects like adaptation and color appearance. Further, [6] presents a method based on logarithmic compression of luminance values, imitating the human response to light. Recently, [7] formulates the tone mapping problem as a quantization process and employs an adaptive conscience learning strategy to obtain mapped images. Perhaps the most comprehensive technique is still that of [8], which first improves histogram equalization and then extends this idea to incorporate models of human contrast sensitivity, glare, spatial acuity, and color sensitivity effects.

Local tone mapping techniques use spatially varying mapping functions. [9-12] are based on the same principle of decomposing an image into layers and differently compressing them. Usually, layers with large features are strongly compressed to reduce the dynamic range while layers of details are untouched or even enhanced to preserve details. [13] presents a method based on a multiscale version of the Retinex theory of color vision. [14] attempts to incorporate traditional photographic techniques to the digital domain for reproducing HDR images. [15] compresses dynamic range through the manipulation of the gradient domain in the logarithmic space. More recently, [16] domain in the logarithmic space. While local histogram, proposes a novel method by adjusting the local histogram. There has been little published research to explore rendering HDR images on the GPU [17,18] and these wor

After global thresholding

Background – original image

