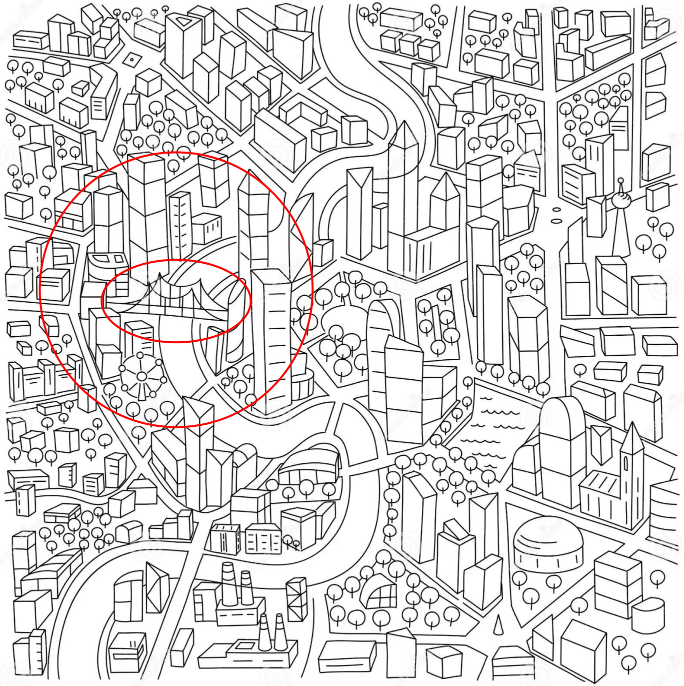
Neighborhood Filters

Section 2.6, Book 1, The ITK Software Guide

Saeedha Nazar (200417M)
BM4302 Medical Image Processing
Department of Electronic & Telecommunication Engineering
University of Moratuwa

Introduction to Neighborhood Filters

Concept of Locality

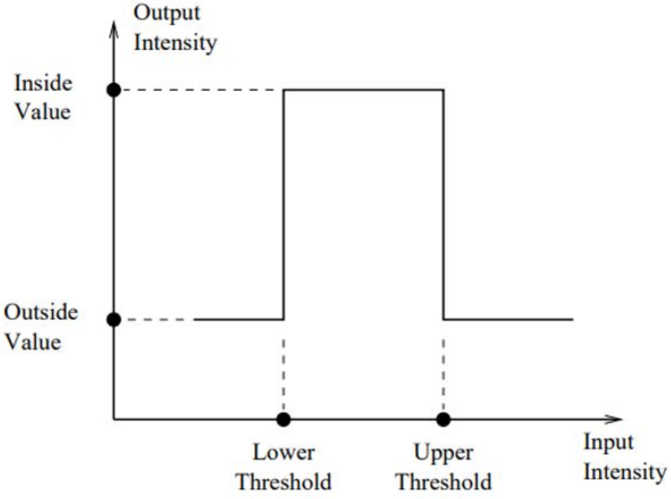


Source:
https://cartoondealer.com/image/214356415/locality-area-map-sketch-city-village-map-board-game-hand-drawn-vector-line-open-paths-editable-outline.html

Introduction to Neighborhood Filters

Previously

Intensity Transformations

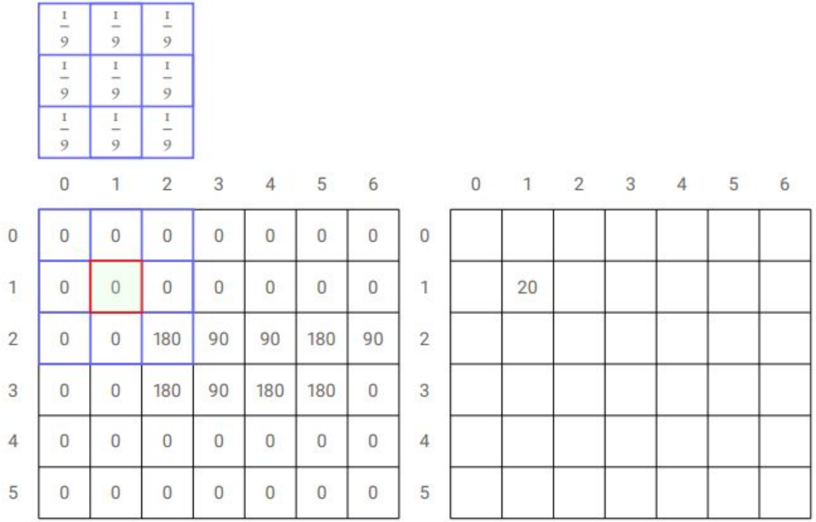


Source: Thresholding, Kalinga

Output pixel depends on input pixel

Spatial Filters

3×3 filter in 2D images (2D convolution)



$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
2	0	0	180	90	90	180	90
3	0	0	180	90	180	180	0
4	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0

	0	1	2	3	4	5	6
0							
1		20					
2							
3							
4							
5							

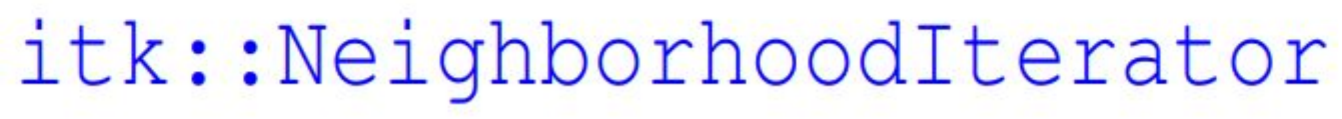
Source: EN3160 Fundamentals of Image Processing and Machine Vision: Spatial Filtering, Dr. Ranga Rodrigo

Output pixel depends on the neighborhood of the input pixel

Introduction to Neighborhood Filters

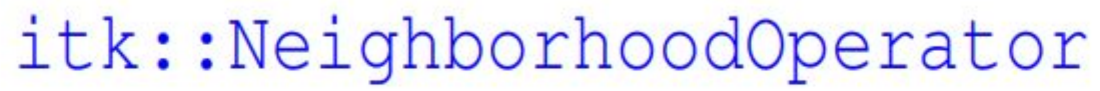
ITK’s Approach

- Filters work on images of any dimension due to the internal use of following classes



```
itk::NeighborhoodIterator
```

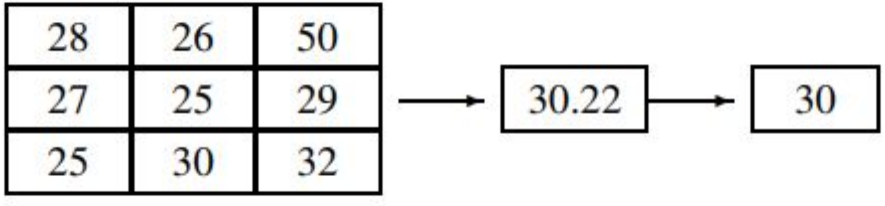
Iterates over image pixels and accesses local neighborhoods



```
itk::NeighborhoodOperator
```

Mean Filter

- Known as box filter in image processing
- Sensitive to outliers



28	26	50
27	25	29
25	30	32

output pixel = statistical mean of the neighborhood(input pixel)

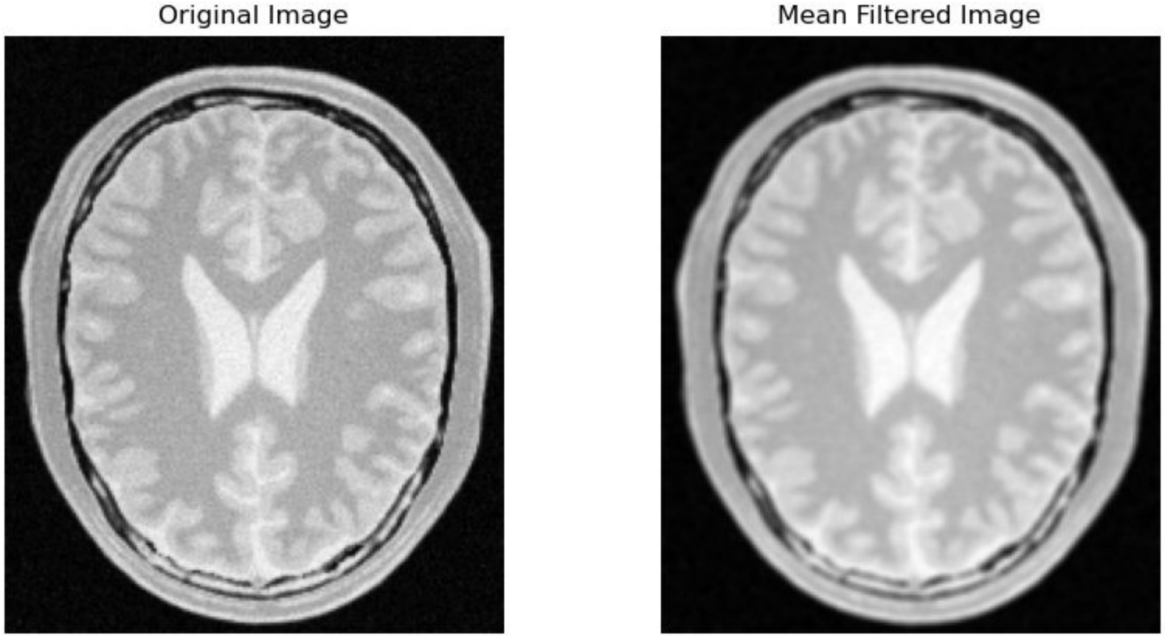
- Handling Border Pixels in IT

Default Behavior - ZeroFluxNeumannBoundaryCondition

(Values of pixels outside the image boundary are
equal to the nearest pixel value inside the boundary)

Mean Filter

- Edges are rapidly degraded by the diffusion of intensity values among neighbors.



Original Image

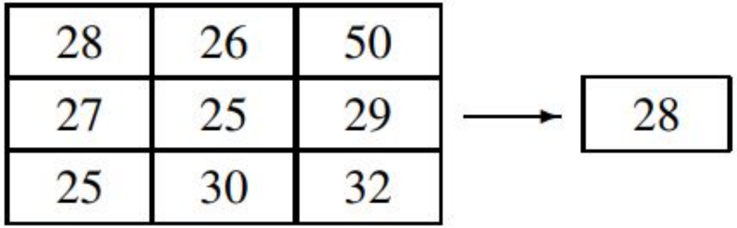
Mean Filtered Image

Source image: MRI proton density brain image, The ITK Software Guide

Mean Filter Code

Median Filter

- Used as a robust approach for noise reduction
- Especially effective against salt-and-pepper noise
- Robust to outliers

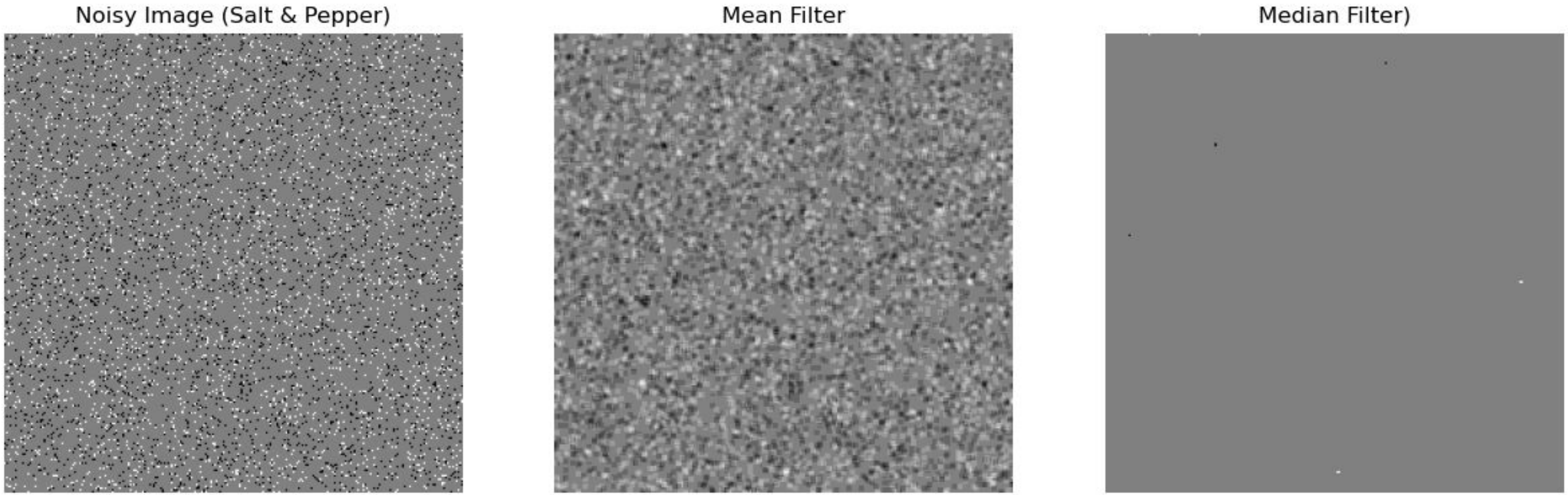


28	26	50
27	25	29
25	30	32

→ 28

output pixel = statistical median of the neighborhood(input pixel)

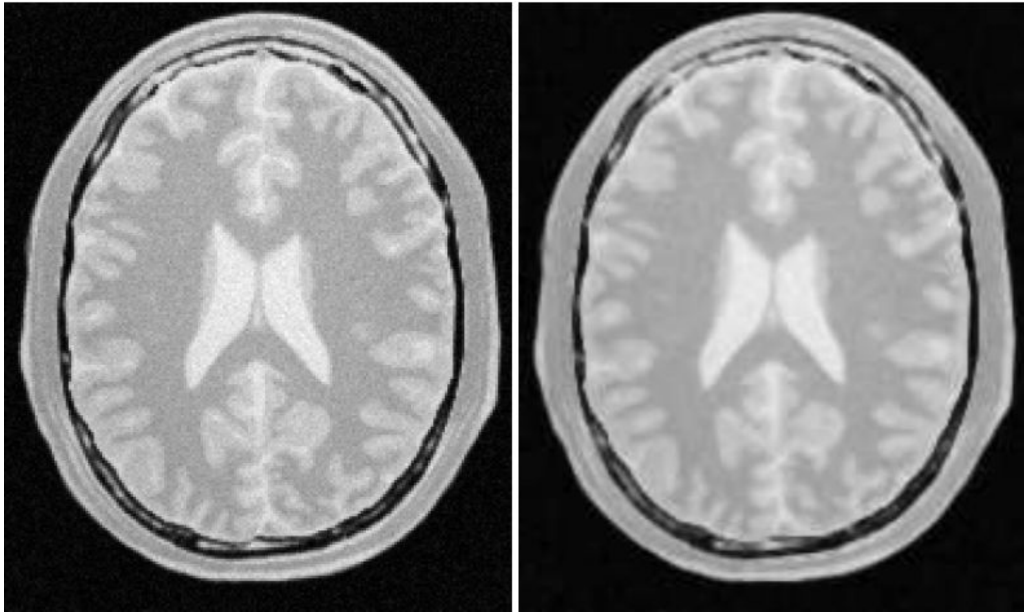
Median Filter



- Works on images of any dimension due to the internal use of itk::NeighborhoodIterator and itk::NeighborhoodOperator

Median Filter

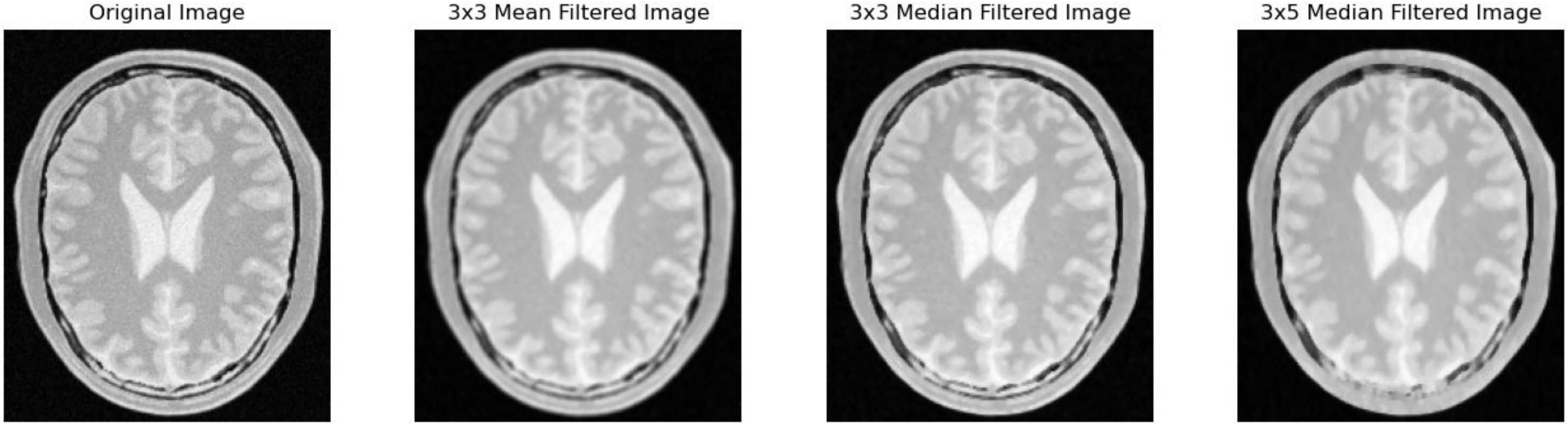
- Ability to reduce noise while **moderately** preserving edges in the image



Source image: MRI proton density brain image, The ITK Software Guide

Median Filter Code

So far…



Original Image

3x3 Mean Filtered Image

3x3 Median Filtered Image

3x5 Median Filtered Image

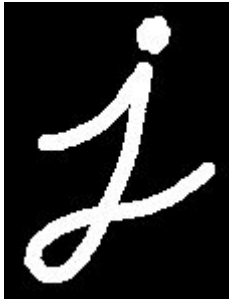
Mathematical Morphology

- Used to extract image components useful for representation and description

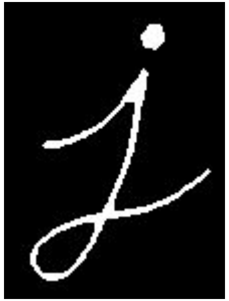
- Two types of image morphology algorithms:
  a. Filters that operate on **binary** images
  b. Filters that operate on **grayscale** images

Binary Filters

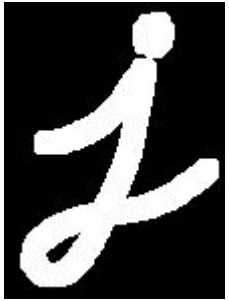
- Basic Operations:
  a. **Erosion**: Shrinks the objects in a binary image by eroding away the boundaries. Used to remove small spurious details or erode away small bridges between objects.
  b. **Dilation**: Expands the objects in a binary image by adding pixels to the boundaries. Used to close small holes within objects or to join adjacent objects.



original



Erosion
(shrink)

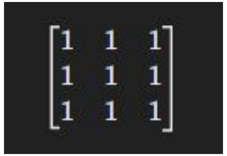


Dilation
(grow)

Source: OpenCV: Eroding and Dilating

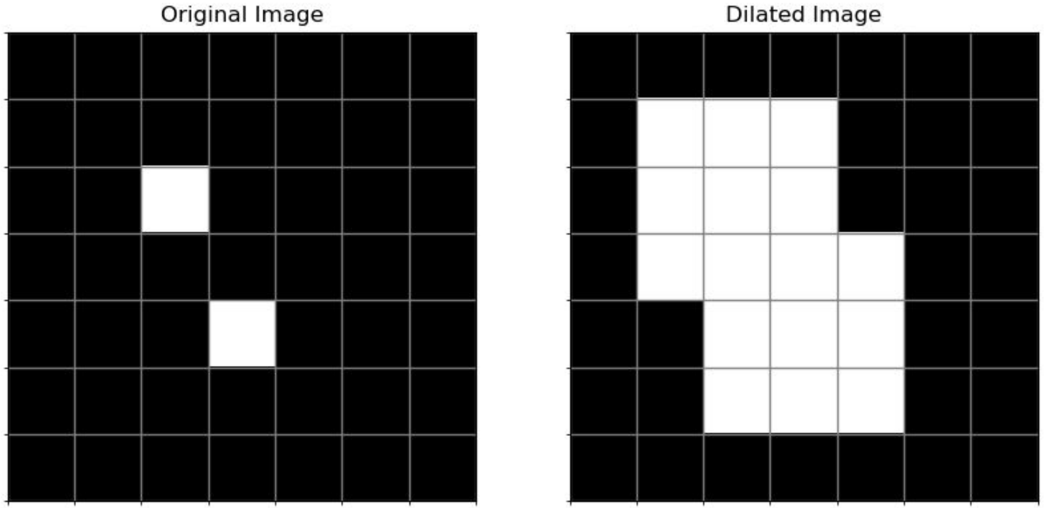
Binary Filters

- **Structuring Element**: These operations use a predefined shape (like a disk, square, etc.) that determines the neighborhood of each pixel. The structuring element "slides" over the image to apply the morphological operations.



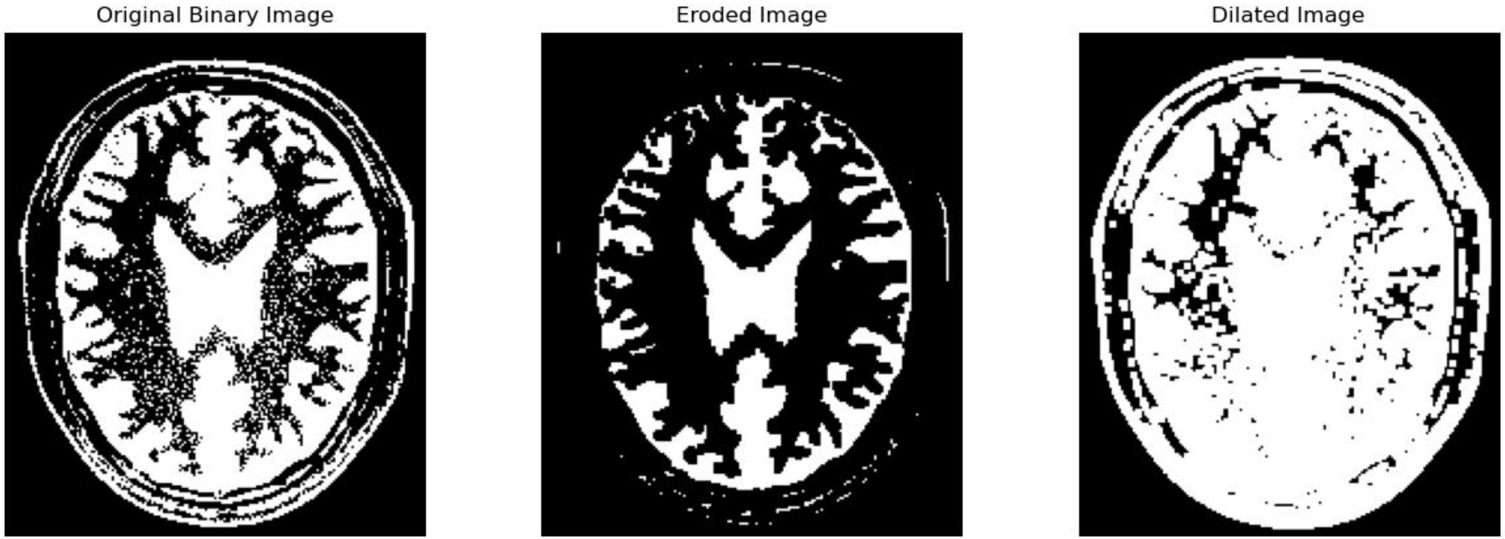
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Structuring Element



Binary Filters

3x3 ball-shape Structuring Element



Original Binary Image

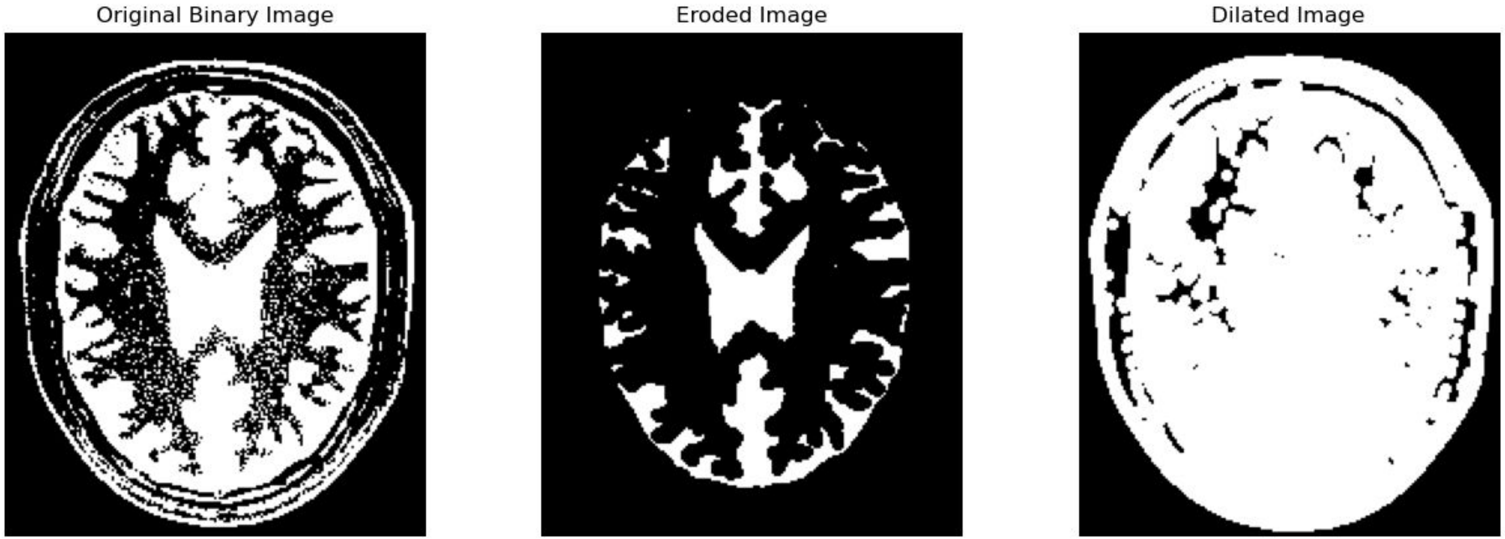
Eroded Image

Dilated Image

Source image: MRI proton density brain image, The ITK Software Guide

Binary Filters

3x3 ball-shape Structuring Element



Source image: MRI proton density brain image, The ITK Software Guide

Binary Filter Code

Grayscale Filters

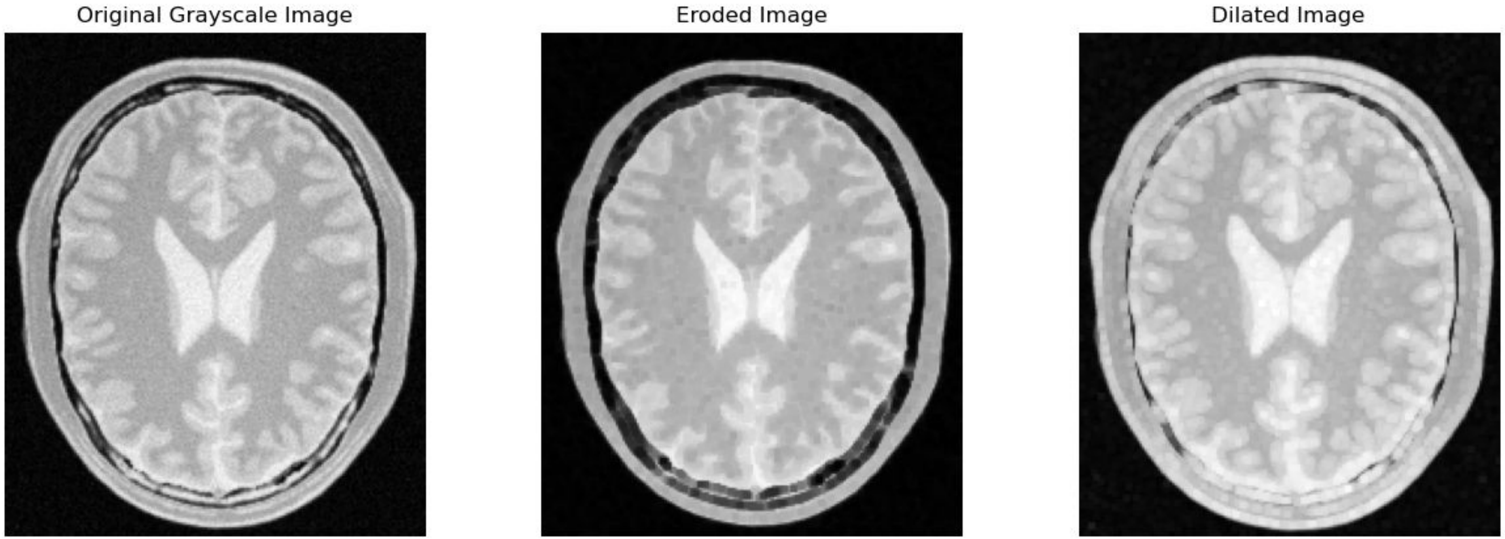
- Grayscale morphological operations extend these concepts to grayscale images, allowing for more nuanced processing.

- Basic Operations:
  a. **Erosion**
     - Reduces the intensity of pixels, making bright regions smaller
     - Finds the minimum value within the neighborhood defined by a structuring element and assigns to the central pixel

  b. **Dilation**
     - Increases the intensity of pixels, expanding bright regions
     - Finds the maximum value within the neighborhood defined by a structuring element and assigns to the central pixel

Grayscale Filters

3x3 ball-shape Structuring Element



Original Grayscale Image

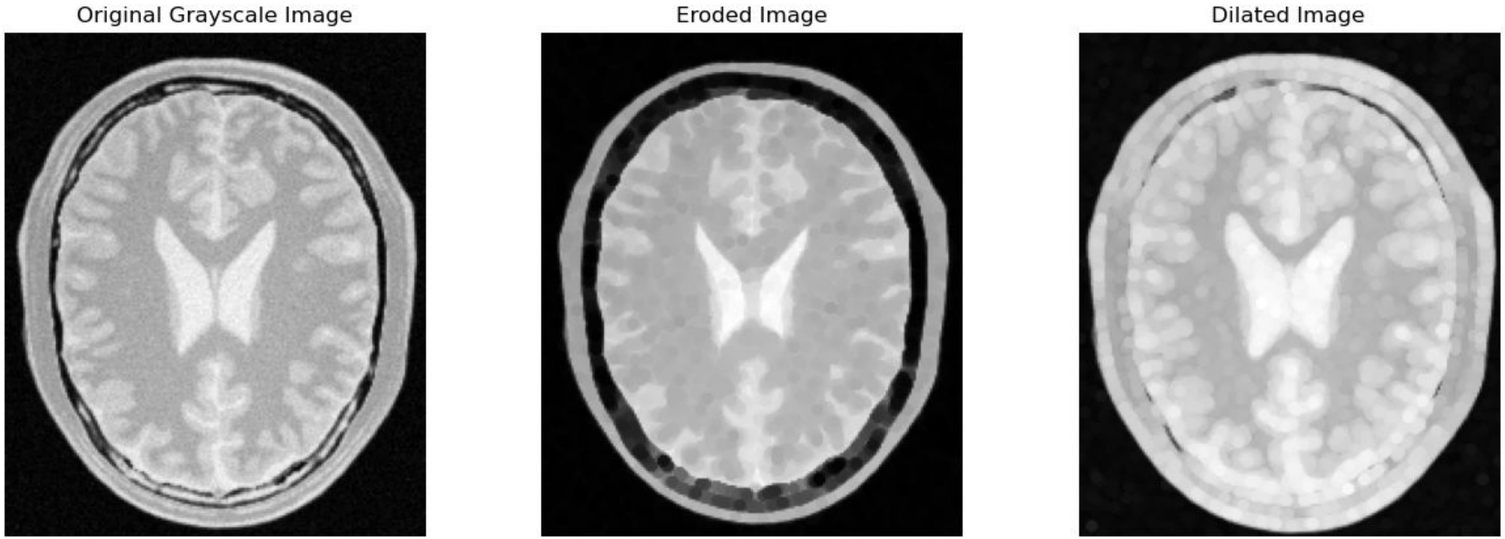
Eroded Image

Dilated Image

Source image: MRI proton density brain image, The ITK Software Guide

Grayscale Filters

5x5 ball-shape Structuring Element



Original Grayscale Image

Eroded Image

Dilated Image

Source image: MRI proton density brain image, The ITK Software Guide

Grayscale Filter Code

Voting Filters

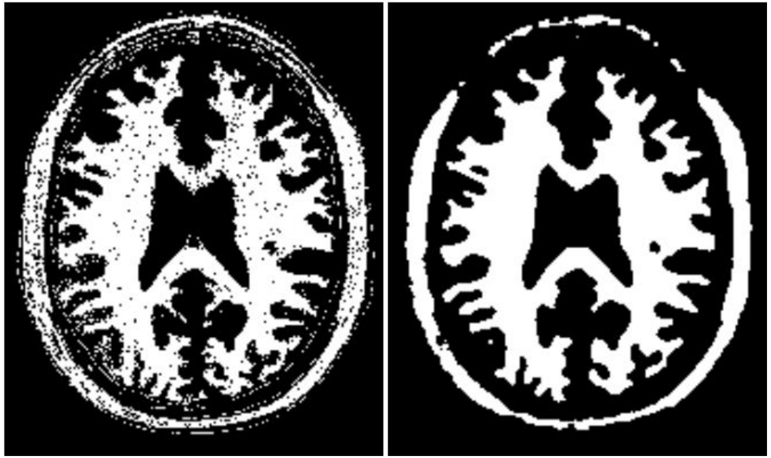
- Voting filters are a generic family of filters
- Both the Dilate and Erode filters from Mathematical Morphology are particular cases of the broader family of voting filters
- Value of each pixel is determined by counting the values of neighboring pixels and applying a rule based on the count

- E.g.,
  a. **Erosion**: A pixel is eroded (turned off) if a sufficient number of its neighbors are already off (background)
  b. **Dilation**: A pixel is dilated (turned on) if a sufficient number of its neighbors are on (foreground).

Binary Median Filter `itk::BinaryMedianImageFilter`

- A specific case of a voting filter
- Functions similarly to a **median filter** but is optimized for **binary** images
- Counts the number of foreground (on) and background (off) pixels in the neighborhood and sets the pixel value based on the **majority**
- Used for **noise reduction** in binary images. It can **smooth** contours while preserving the overall shape of objects in the image
- Dramatic reduction in **impulse noise spikes**

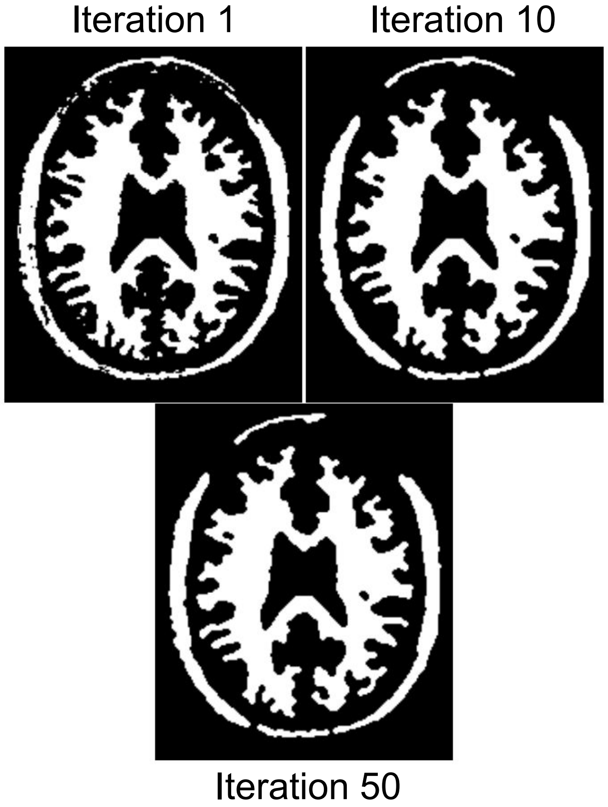
radius of 2, a 5×5 classical neighborhood



Source image: MRI proton density brain image, The ITK Software Guide

Binary Median Filter

- Repeated application of the median filter can progressively smooth the image
- With each iteration, noise (particularly salt-and-pepper noise) is reduced, and the image becomes smoother.
- However, as the number of iterations increases, not only is the noise reduced, but also **finer details in the image may be blurred** out or lost, leading to a loss of sharpness.
- There is a **tradeoff** in **noise reduction and the sharpness**


Iteration 1

Iteration 10

Iteration 50

All followed a radius of 1, 3x3 window

Source image: MRI proton density brain image, The ITK Software Guide

Hole Filling Filter

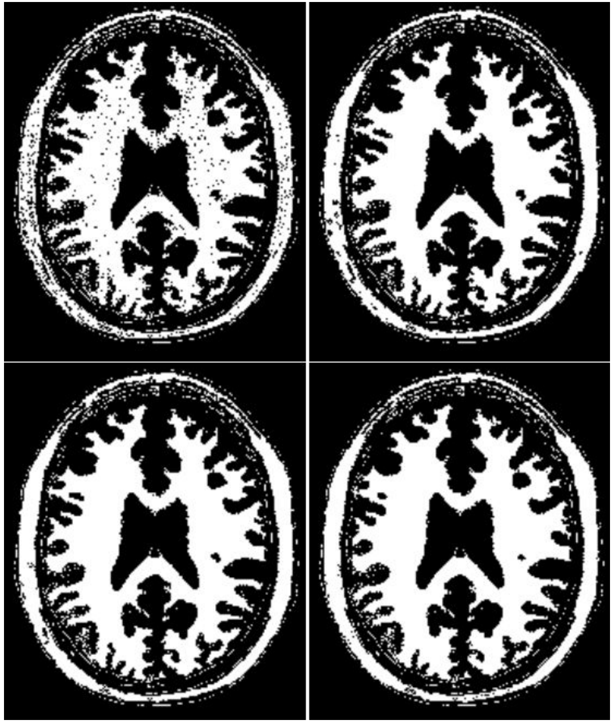
- A specialized type of voting filter
- Designed to fill "**holes**" or cavities in binary images. A hole is defined as a region of background pixels (0) that is surrounded by foreground pixels (255)

- **Voting Mechanism**: The filter determines whether to convert a background pixel into a foreground pixel based on the majority of its neighbors. If the majority of the surrounding pixels are foreground, the central background pixel is converted to foreground. This process is controlled by the **Majority Threshold**

Voting Binary Hole Filling Filter

```
itk::VotingBinaryHoleFillingImageFilter
```

- Useful for filling in small holes in segmented binary images, smoothing contours, and ensuring more uniform regions
- The filter is flexible, allowing the user to specify the neighborhood size and the majority threshold to control how aggressively holes are filled
- Reduces noise **both in the background and foreground** of the image
- The VotingBinaryHoleFillingImageFilter is configured with:
    a. **Radius**: The neighborhood radius is set to define the area around each pixel that will be considered when deciding whether to fill a hole.
    b. **Background and Foreground Values**: These are set to 0 and 255, respectively, to specify what constitutes the background and foreground in the binary image.
    c. **Majority Threshold**: This is the number of foreground pixels required to change a background pixel to foreground.

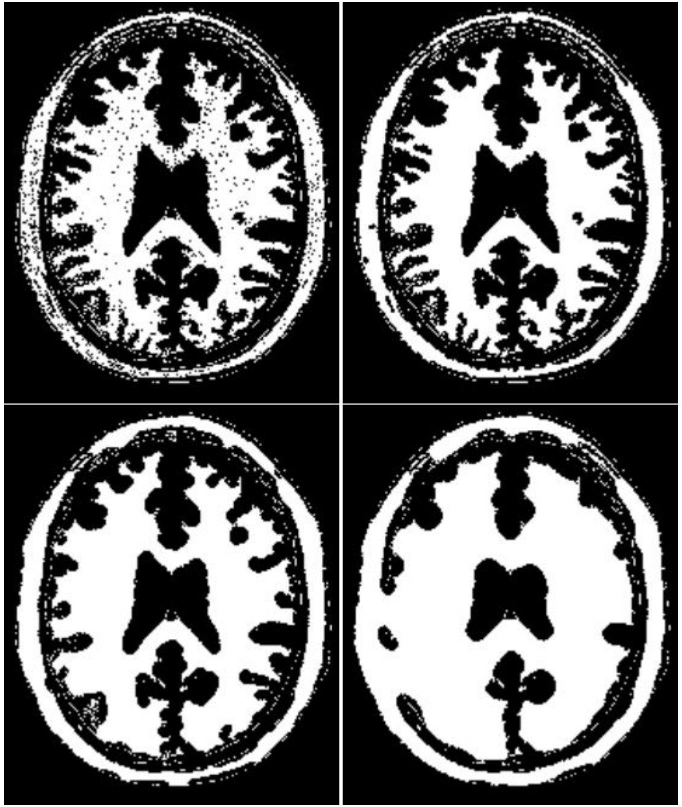


A thresholded slice of MRI brain image using neighborhood radii of 1,1, 2,2 and 3,3 that correspond respectively to neighborhoods of size 3 × 3, 5 × 5, 7 × 7

Iterative Hole Filling

`itk::VotingBinaryIterativeHoleFillingImageFilter`

- Applies the hole filling process repeatedly until **no more changes** occur or a **specified number of iterations** is reached
- This approach is particularly effective for filling **large holes** or **cavities** that require more than one pass to fill completely
- **Smaller Neighborhoods (e.g., 3x3)**: More localized, sensitive to small features, better at preserving fine details, but might not smooth out larger noise areas effectively.
- **Larger Neighborhoods (e.g., 7x7)**: Smoother results, better at reducing larger noise regions and filling large holes, but can cause more blurring and loss of fine details.



VotingBinaryIterativeHoleFillingImageFilter filter on a thresholded slice of MRI brain image using neighborhood radii of 1,1, 2,2 and 3,3 that correspond respectively to neighborhoods of size 3×3, 5×5, 7×7

Applications of Neighborhood Filters

- Feature Extraction
    - Removing noise
    - Smoothing edges
    - Enhancing or suppressing features in grayscale images (**Morphological filters**)
- Noise Reduction
    - **Median Filter (MRI/CT scans)**
    - **Mean Filter**
- Image Restoration and Inpainting
    - **Hole Filling Filters**
    - **Neighborhood filters**
- Image Registration
    - **Smoothing Filters**: Used in pre-processing steps to reduce noise and enhance image features before registration. Image registration is crucial in aligning images from different modalities (e.g., CT and MRI).
    - **Edge-Preserving Filters**
- Segmentation
    - **Morphological Operations (Erosion, Dilation, Opening, Closing) are essential in segmenting different structures in medical images**