THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Image Filtering

Image filtering is used to:

- Remove noise
- Sharpen contrast
- Highlight contours
- Detect edges
- Other uses?

Image filters can be classified as linear or nonlinear.

Linear filters are also know as convolution filters as they can be represented using a matrix multiplication.

Thresholding and image equalisation are examples of nonlinear operations, as is the median filter.

1

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Median Filtering

Median filtering is a nonlinear method used to remove noise from images.

It is widely used as it is very effective at removing noise while preserving edges.

It is particularly effective at removing ‘salt and pepper’ type noise.

The median filter works by moving through the image pixel by pixel, replacing each value with the median value of neighbouring pixels.

The pattern of neighbours is called the "window", which slides, pixel by pixel, over the entire image.

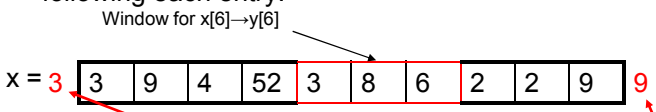
The median is calculated by first sorting all the pixel values from the window into numerical order, and then replacing the pixel being considered with the middle (median) pixel value.

2

Median Filtering example

The following example shows the application of a median filter to a simple one dimensional signal.

A window size of three is used, with one entry immediately preceding and following each entry.



y[1] = median[3 3 9] = 3
y[2] = median[3 4 9] = 4
y[3] = median[4 9 52] = 9
y[4] = median[3 4 52] = 4
y[5] = median[3 8 52] = 8

y[6] = median[3 6 8] = 6
y[7] = median[2 6 8] = 6
y[8] = median[2 2 6] = 2
y[9] = median[2 2 9] = 2
y[10] = median[2 9 9] = 9

y = 3 4 9 4 8 6 6 2 2 9

3

For y[1] and y[9], extend the left-most or right most value outside the boundaries of the image
same as leaving left-most or right most value unchanged after 1-D median

Median Filtering

In the previous example, because there is no entry preceding the first value, the first value is repeated (as is the last value) to obtain enough entries to fill the window.

What effect does this have on the boundary values?

There are other approaches that have different properties that might be preferred in particular circumstances:

- Avoid processing the boundaries, with or without cropping the signal or image boundary afterwards.
- Fetching entries from other places in the signal. With images for example, entries from the far horizontal or vertical boundary might be selected.
- Shrinking the window near the boundaries, so that every window is full.

What effects might these approaches have on the boundary values?

4

Median Filtering

On the left is an image containing a significant amount of salt and pepper noise. On the right is the same image after processing with a median filter.

Notice the well preserved edges in the image.

There is some remaining noise on the boundary of the image. Why is this?

5

Median Filtering example 2

2D Median filtering example using a 3 x 3 sampling window:

Keeping border values unchanged

Sorted: 0,0,1,1,1,2,2,4,4

Input

1	4	0	1	3	1
2	2	4	2	2	3
1	0	1	0	1	0
1	2	1	0	2	2
2	5	3	1	2	5
1	1	4	2	3	0

Output

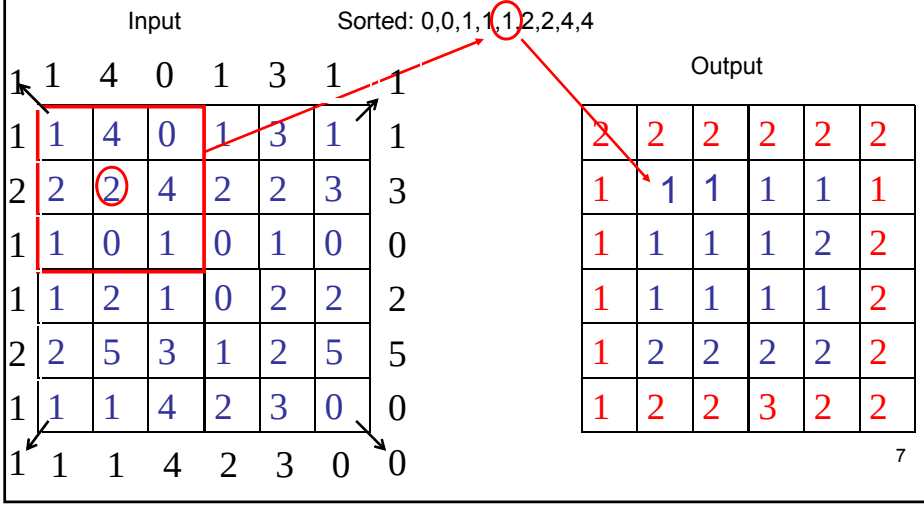
1	4	0	1	3	1
2	1	1	1	1	3
1	1	1	1	2	0
1	1	1	1	1	2
2	2	2	2	2	5
1	1	4	2	3	0

6

Median Filtering - Boundaries

2D Median filtering example using a 3 x 3 sampling window:

Extending border values outside with values at boundary

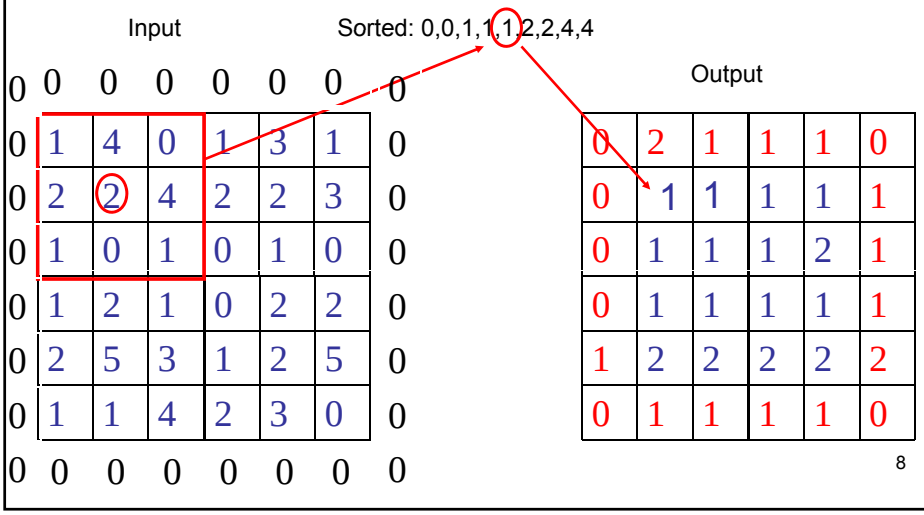


7

Median Filtering - Boundaries

2D Median filtering example using a 3 x 3 sampling window:

Extending border values outside with 0s



8

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Average Filtering

Average (or mean) filtering is a method of 'smoothing' images by reducing the amount of intensity variation between neighbouring pixels.

The average filter works by moving through the image pixel by pixel, replacing each value with the average value of neighbouring pixels, including itself.

There are some potential problems:

- A single pixel with a very unrepresentative value can significantly affect the average value of all the pixels in its neighbourhood.
- When the filter neighbourhood straddles an edge, the filter will interpolate new values for pixels on the edge and so will blur that edge. This may be a problem if sharp edges are required in the output.

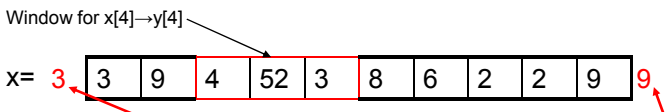
9

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Average Filtering

The following example shows the application of an average filter to a simple one dimensional signal.

A window size of three is used, with one entry immediately preceding and following each entry.



y[1] = round((3+3+9)/3)= 5
y[2] = round((3+9+4)/3)= 5
y[3] = round((9+4+52)/3)= 22
y[4] = round((4+52+3)/3)= 20
y[5] = round((52+3+8)/3)= 21

y[6] = round((3+8+6)/3)= 6
y[7] = round((8+6+2)/3)= 5
y[8] = round((6+2+2)/3)= 3
y[9] = round((2+2+9)/3)= 4
y[10] = round((2+9+9)/3)= 7

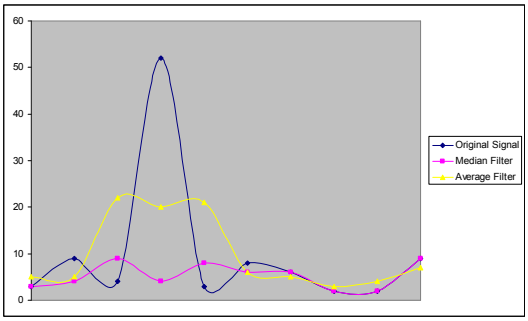
y= 5 5 22 20 21 6 5 3 4 7

For y[1] and y[9], extend the left-most or right most value outside the boundaries of the image

10

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Filter Comparison



The graph above shows the 1D signals from the median and average filter examples.

What are the differences in the way the filters have modified the original signal?

11

3 by 3 Average filtering

- Consider the following 3 by 3 average filter:
- We can write it mathematically as:

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

$$I\_new(x,y) = \sum_{j=-1}^{1}\sum_{i=-1}^{1} 1 \times I\_old(x+i, y+j)$$

$$I\_new\_normalized(x,y) = \frac{1}{\sum_{j=-1}^{1}\sum_{i=-1}^{1} 1} \sum_{j=-1}^{1}\sum_{i=-1}^{1} 1 \times I\_old(x+i, y+j)$$

- Why normalizing is important ?
  - To keep the image pixel values between 0 and 255

12

Average Filtering example 2

2D Average filtering example using a 3 x 3 sampling window:

Keeping border values unchanged

Average = round(1+4+0+2+2+4+1+0+1)/9 = 2

Input

1	4	0	1	3	1
2	2	4	2	2	3
1	0	1	0	1	0
1	2	1	0	2	2
2	5	3	1	2	5
1	1	4	2	3	0

Output

1	4	0	1	3	1
2	2	2	2	1	3
1	2	1	1	1	0
1	2	1	1	1	2
2	2	2	2	2	5
1	1	4	2	3	0

13

Average Filtering - Boundaries

2D Average filtering example using a 3 x 3 sampling window:

Extending border values outside with values at boundary

Input

Average = round(1+4+0+1+4+0+2+2+4)/9 = 2

1	1	4	0	1	3	1
1	1	4	0	1	3	1
2	2	2	4	2	2	3
1	1	0	1	0	1	0
1	1	2	1	0	2	2
2	2	5	3	1	2	5
1	1	1	4	2	3	0
1	1	1	4	2	3	0

Output

2	2	2	2	2	2
2	2	2	2	1	2
1	2	1	1	1	2
2	2	1	1	1	2
2	2	2	2	2	2
2	2	3	3	2	2

14

Average Filtering - Boundaries

2D Median filtering example using a 3 x 3 sampling window:

Extending border values outside with 0s (Zero-padding)

Input

Average = round(2+5+0+3+0+0+0+0+0)/9 = 1

0	0	0	0	0	0	0	0
0	1	4	0	1	3	1	0
0	2	2	4	2	2	3	0
0	1	0	1	0	1	0	0
0	1	2	1	0	2	2	0
0	2	5	3	1	2	5	0
0	1	1	4	2	3	0	0
0	0	0	0	0	0	0	0

Output

1	1	1	1	1	1
1	2	2	2	1	1
1	2	1	1	1	1
1	2	1	1	1	1
1	2	2	2	2	2
1	2	2	2	1	1

15

Average Filtering

On the left is an image containing a significant amount of salt and pepper noise. On the right is the same image after processing with an Average filter.

What are the differences in the result compared with the Median filter?

Is this a linear (convolution) or nonlinear filter?

16