Page No.
Date

23/10/20

Edge Detection

* Edge detion can be done in two ways: Vertical and Horizontal edge ~~detion~~ detection.

* Suppose we take a complicate (with many objects and crevices) image which is grayscaled.

* So the output of two detection methods would be two pics with only vertical edges highlighted and on the other horizontal edges.

Vertical Edge Detection

Suppose we take a 6x6 ~~image~~ image, as it is a grayscale image, so represented as 6x6x1.

$3\times1+1\times1+2\times1+0\times0+5\times0+7\times0+1\times-1+8\times-1+2\times-1=-5$

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

so filter is slided over the image

$*$ ↓ convolutional

1	0	-1
1	0	-1
1	0	-1

3x3 filter

6x6

4x4 image

Page No.
Date

So the o/p would be:

-5	-4	0	8
-10	-2	2	3
0	-2	-4	-7
-3	-2	-3	-16

The filter is called a vertical edge detector.

Why this filter is called vertical edge detector?

Let's take another example of following grayscale image:

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0

6x6

White	Gray	

(assumption)

But clearly a strong vertical edge in the middle.

×

1	0	-1
1	0	-1
1	0	-1

3x3

White	Gray	Black

Lower the number darker the shade (assumption)

The output would be:

0	30	30	0
0	30	30	0
0	30	30	0
0	30	30	0
Gray	white		Gray

The dimension may seem wrong as the image is very small, hence, this

So, the vertical filter can be seen as very light pixel on the left and very dark pixels on the right. The center being '0' is not the important.

⇒ Now, if we reverse the image, that is rotate in 180° clockwise around the centre elements.

The new output would be:

0	-30	-30	0
0	-30	-30	0
0	-30	-30	0
0	-30	-30	0
Gray	Black		Gray

This shows that the transition is dark to light, not light to dark.

Page No.
Date

Now we know:

1	0	-1
1	0	-1
1	0	-1

is a vertical filter

So,

1	1	1
0	0	0
-1	-1	-1

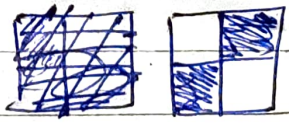
is a horizontal filter where the upper part is brighter and below is darker.

So, now, if we take:

10	10	10	0	0	0
10	10	10	0	0	0
10	10	10	0	0	0
0	0	0	10	10	10
0	0	0	10	10	10
0	0	0	10	10	10

$*$

1	1	1
0	0	0
-1	-1	-1

⇒ Basically 

o/p.

0	0	0	0
30	10	-10	-30
30	10	-10	-30
0	0	0	0

Upper Brighter lower darker ←

→ Lower brighter upper darker

The filters above is just one kind of edge detection filter and there are other types too.

Like for vertical edge detection:

1	0	-1
2	0	-2
1	0	-1

Sobel filter (more robust than simple filter)

another example

3	0	-3
10	0	-10
3	0	-3

Scharr filter

In deep learning, we can put the filter elements as simple as parameters that are changed through backpropagation. Not only for edges straight line image but also edges with different angles. Hence, we leave it to the neural n/w.

Blob Detection

* What is a BLOB?

Blobs are bright on dark or dark on bright regions in an image. A blob is a group of connected pixels in an image that share some common property (eg. grayscale value). In the image above, the dark connected regions are blobs, and the goal

Page No. _____
Date

of blob detection is to identify and mark these regions.

★ How does it work?

1. Thresholding: Convert the source images to several binary images by thresholding the source image with thresholds. ~~Starting at~~ These thresholds are incremented.

2. Grouping: In each binary image, connected white pixels are grouped together.

3. Merging: The centers of the binary blobs in the binary images are computed, and blobs located closer than a minimum distance are merged.

4. Center & Radius Calculation: The centers and radii of the new merged blobs are computed and returned.