Research Article

Euclidean Distance Filter for Image Processing

Hamid Hassanpour*

B.Sc. degree from Amirkabir University of Technology.

Corresponding Author: Hamid Hassanpour, B.Sc. degree from Amirkabir University of Technology

Abstract

In this paper, we illustrate concept of Euclidean distance filter and its application in image processing. We will examine mathematical concepts and illustrate examples to show practical results of applying these filters on images.

Keywords: Image Processing, Image Filters, Euclidean Distance.

1. Introduction

Image filters have been used and known for many years and their application is evidence of their importance in enhancing, edge detection and noise reduction of images. These filters are examined and analysed both in time and frequency domains. General framework of image filters is mathematical calculations, especially probabilistic methods, on pixel values. Geometrical filters are missing components of image processing and as today, these is no analysis and contribution in this area. The key concept of geometrical filters is considering geometrical position of each pixel respect to a defined coordinate. In this way each pixel has both pixel value and weight. This weight can be defined by geometrical concepts. As an example, in Euclidean distance filter, every pixel's weight is a function of its Euclidean distance from centre of the filter. In this way, each pixel's weight within the filter range contributes in determining pixel value locating in centre of the area that the filter is applied.

2. Methodology

Euclidean distance filter is a rectangular filter where value of each block is its Euclidean distance from centre of the filter. It is assumed that position of each filter's block can be pointed by x-y coordinates, where (0,0) coordinate is position of centre block of the filter. Filter value at each block can be shown by F(x,y). Figure 1 illustrates these concepts. Figure a is a is a 9*9 filter and it shows position of each filter's block in x-y coordinate and its value F(x,y). As a result, each block's value is its Euclidean distance from centre of the filter, which is calculated inside filter's block in Figure b.
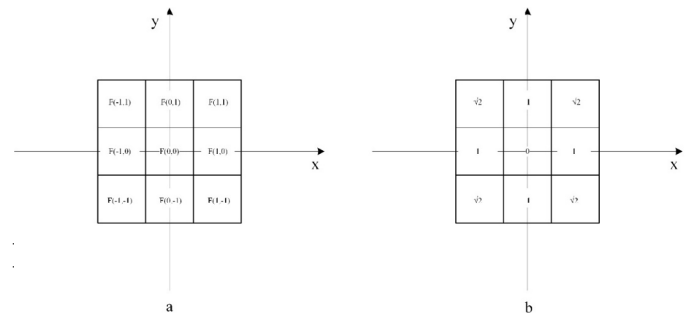
Figure 1: Basic Concepts of Euclidean Distance Filter, x-y Coordinate Position of Each Block (a), and Geometric Distance of Each Block (b).

Figure 2 illustrates method of applying Euclidean filter to an image. The right shape is specific area of an image that Euclidean distance filter is applying. Figure a is a rectangle area of that image and each pixel is coordinated respect to center of the filter, resulting having of X(x,y). Figure b is a 9*9 filter consisting of its values and the multiplication (*) is applied element-wise. The result of applying the filter on pixel values, is a 9*9 block consisting of product of each pixel value and its Euclidean distance from centre of the filter. For calculating pixel value that is coincided with center of the filter, which in this case is pixel (0,0), these products can be summed up and averaged.

X(-1,1)	X(0,1)	X(1,1)
X(-1,0)	X(1,0)	X(1,0)
X(-1,-1)	X(0,-1)	X(1,-1)

*

√2	1	√2
1	0	1
√2	1	√2

Figure 2: Multiplication of Pixel Values (left) With Filter Values (right).

In order to reduce noise of an image and eliminate sharp areas, inverse value of Euclidean distance can be used, i.e., as the pixel is further away from center of the filter, it's contribution in calculating pixel value that coincide with center of the filter is decreased. Mathematics of calculating filter's values is illustrated as below:

$$D(x,y) = \sqrt{x^2 + y^2} \tag{1}$$

$$S(x,y) = \frac{1}{D(x,y)} \tag{2}$$

$$S(0,0) = \gamma \tag{3}$$

$$F(x,y) = \frac{S(x,y)}{\sum_{x,y \in n} S(x,y)} \tag{4}$$

where $D(x,y)$ is Euclidean distance of each filter's block from center of the filter, $S(x,y)$ is inverse of Euclidean distance for noise reduction and smoothing, and $\gamma\gamma$ is contribution factor of center of the filter. As $\gamma\gamma$ increases contribution of center of the filter increases and result picture is more similar to original one. On the other hand, if $\gamma\gamma$ decreases, center's contribution decrease and the result picture will become more smoothed. Filter values,

(x, y), are normalized values of $S(x, y)$. In this way output pixel value (x, y) is calculated using equation below:
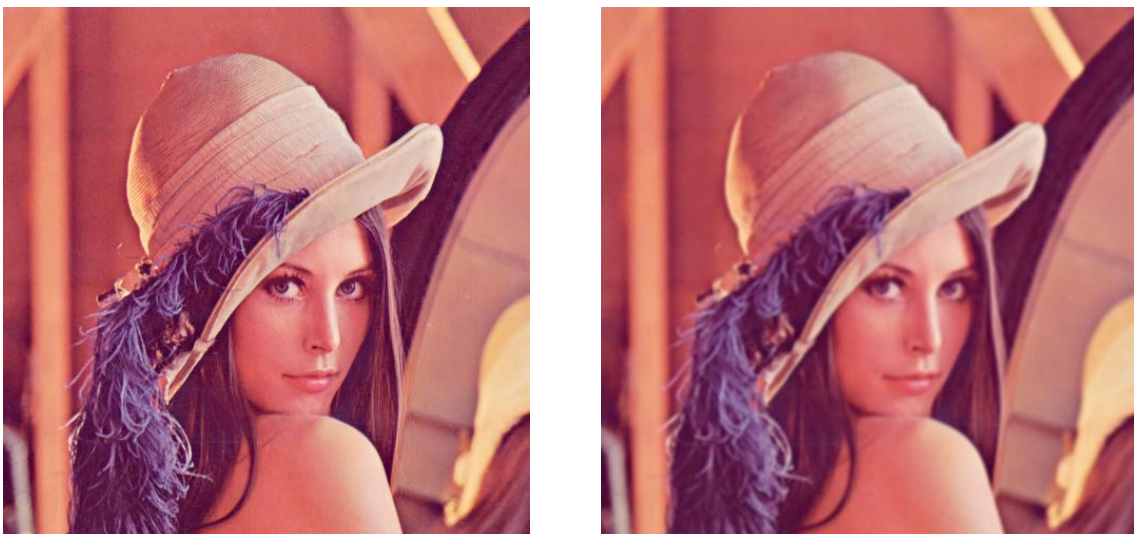
$$Y(x,y) = \sum_{x,y \in n} F(x,y) * X(x,y) \tag{5}$$

where $*$ is element-wise product of filter $F(x, y)$ and input pixel $X(x, y)$.

3. Results

Figure 3: Original Image (left) and Filtered Image (right).

Figure 4: Original Image (left) and Filtered Image (right).

Figure 3 and Figure 4 illustrate effect of applying Euclidean distance filter to an original image that is implemented in Euclidean distance filter implementation (2021). To both images, a 5*5 filter is applied, and as it shows, sharp areas of pictures are smoothed.

As the size of filter increases, more neighbour pixels are contributing in calculation of pixel value, and as expected, the picture will become more smoothed. Figure 5 illustrates this concept.

calculation of pixel value coordinated in center of the filter is heavily biased with this block, resulting of having a picture more similar to original one.

Figure 6: Comparing Original Image (a) and Applying Filter with Gamma=0 (b), Gamma=10 (c), and Gamma=100 (d). As Expected, the Picture Gets More Sharper as Gamma Increases.

Figure 5: Comparing Original Image (a) and Applying 5*5 Filter (b), 10*10 Filter (c), and 15*15 Filter (d). As Expected, the Picture Gets More Smoothed as Size of Filter Increases.

For evaluation of importance of gamma $\gamma\gamma$, we apply a 10*10 filter with different $\gamma\gamma$ values. Picture 6 illustrates effect of $\gamma\gamma$ values on performance of the filter. As $\gamma\gamma$ increases, importance of center block of the filter increases and, as result,

References

1. Euclidian distance filter implementation (2021) [online] (Accessed January 29, 2021).
2. Hamid Hassanpour received his B.Sc. degree from Amirkabir University of Technology. He has worked for 5 years in industry as embedded software engineer. His areas of interest include artificial intelligence and design of embedded systems.