SYSC-5104
METHODOLOGIES FOR DISCRETE-EVENT MODELLING AND SIMULATION (FALL 2016)

Assignment2: Mean Filter

Victor Guimaraes da Silva 101022659

Carleton University

Mean Filter implemented on Cell-DEVS.

Cellular automata is very suitable for image processing because pixels can be treated as cells and because of the parallelism capabilities of Cellular automata, which improves computing efficiency. There are many image processing techniques used for solving a variety of problems such as edge detection, noise reduction, etc. In this assignment the mean filter is going to be implemented as a noise reduction technique.

A mean filter is a linear filter that preserves the borders and reduces the noise as well as smoothing the image. It replaces each pixel in the image by the average of all neighbours in the moore neighbourhood. The mean filter is one of the simplest filters and its implementation is very straightforward. Other techniques are able to get better results, but for the sake of simplicity, the mean filter is implemented here just to show the suitability of cellular automata for solving those problems.

The mean filter could be used as a previous stage for edge detection since it smoothes the image reducing the chance of getting wrong edges.

Atomic Model Definition of the Mean Filter.

meanFilter = <X, Y, S, N, d, δ_{int}, δ_{ext}, τ, λ, D>

where

X = Y = $\emptyset$

S = [0,1, ..., 255] (this is the level of gray on the image)

N = { (-1,-1),(-1,0),(0,0),(-1,1),(0,-1),(0,1),(1,-1),(1,0),(1,1)}; (it takes the average of all the neighbours)

d = 100 ms (transport);

τ: $N \rightarrow S$ is defined by the following rule:

S = ((-1,-1)+(-1,0)+(0,0)+(-1,1)+(0,-1)+(0,1)+(1,-1)+(1,0)+(1,1))/9

and the border is nowrapped.

The cell dimension is according to the size of the image.

Results

The following results were obtained from extracting the information of various images using Matlab and printing the value of each pixel as a grayscale image on the .val document. The Matlab script is attached on the project folder.Grayscale images carry only intensity information which means that the RGB values are all the same.

All the images are pictures with salt and pepper noise. The mean filter reduces the amount of noise on the image, however it blurs the image therefore reducing its quality, which is a drawback of this technique. Other kinds of filters present some parameters that assist on controlling the blurring problem.

A limitation on this simulation is the proper set of the simulation time since for this model there is no parameter that is able to indicate when to stop the simulation. For this, 100 ms is enough for a good noise reduction as well as not too much blurring on the image. The simulation showed on figure 3, 4, and 5 was ran for 300 ms. After 100 ms the image is good, but at the end it is completely distorted. The videos generated by the Cell-DEVS visualizer web site are very fast because it takes only one frame to finish the processing. On the web-site it was possible to watch them properly but after downloading them it did not work well. However the images are enough to show the end result.

Figure 1 – Original image with salt and pepper noise.

Figure 2 – Image after processing.

Figure 3 – Original image with salt and pepper noise.

Figure 4 –Image after 100 ms

Figure 5 –Image at the end.

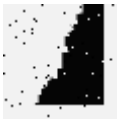

Figure 6 – Original image with salt and pepper noise.

Figure 7 – Image after processing.