Universidad Politécnica de Madrid

Escuela Técnica Superior de Ingenieros Informáticos

Basic Exercises - Theoretical exercises

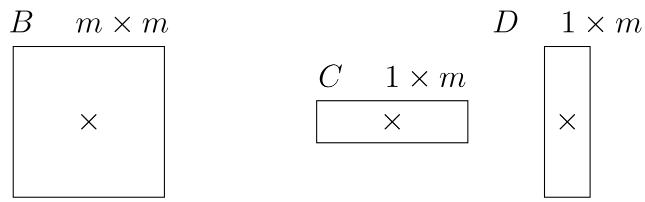
Author: Sergio Marín Sánchez

Email: sergio.marins@alumnos.upm.es

Date: Friday 8th March, 2024

Exercise 2d

Estimate the number of operations of operations to process a $n \times n$ input image using: $\delta_B(I)$ or $\delta_C(\delta_D(I))$.



First of all, it is defined the function $op(f)$ that counts the number of operations required to compute f. It is also defined that computing the maximum of two numbers $a, b \in \mathbb{Z}$ needs only 1 operation, thus, $op(\max\{a, b\}) = 1$.

Using these assumptions computing the maximum of n numbers (x_i) will require $n-1$ operations.

$$op(\underbrace{\max\{\cdots\max\{\max\{}_{\max\ \text{is repeated}\ n-1\ \text{times}}x_1, x_2\}, x_3\}, \cdots, x_n\}) = n-1$$

The number of operations required to compute $\delta_B(I)$ will be computed as the maximum among the structuring element B that has m^2 elements, so it will require m^2-1 operations. This is performed in each pixel of the image, that is n^2 times.

The number of operations required to compute $\delta_C(\delta_D(I))$ will be computed as the maximum among the structuring element C that has m elements, so it will require $m-1$ operations, plus computing the maximum among the other structuring element D that will also require $m-1$ operations because it has m elements too. This is performed in each pixel of the image, that is n^2 times.

In conclusion,

$$n^2(m^2-1) = op(\delta_B(I)) \geq op(\delta_C(\delta_D(I))) = n^2(m-1) + n^2(m-1) = 2n^2(m-1)$$

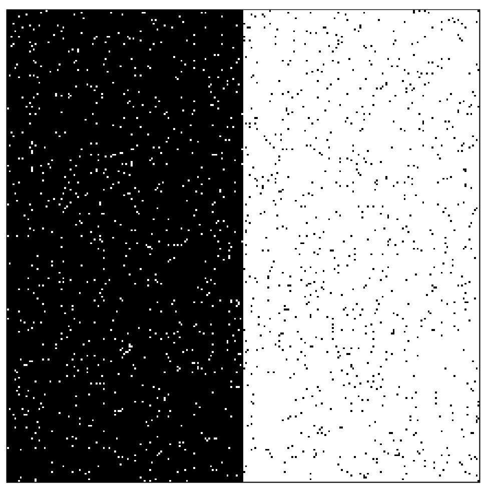
Exercise 8a

Let I be the input image in file `isn_256.pgm`, which has a binary impulsive noise added("salt-and-pepper" noise). Let B be a structuring element square of size 3×3.

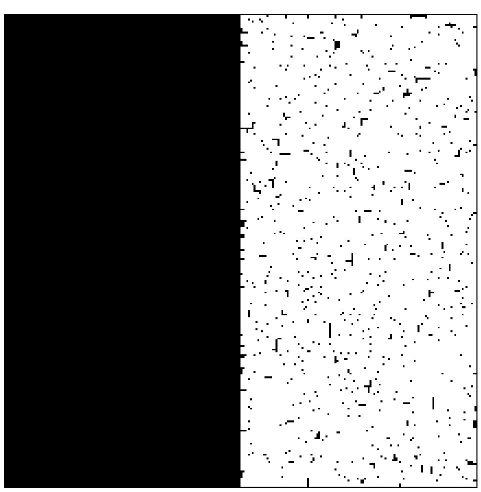
The results of executing the respective filters can be seen in Fig. 1. The best results are obtained using Filter 4 (Opening-Closing $\gamma\varphi$) where it is easy to see there is only one artifact left in the image (on top of the black part).

The second best filter is Filter 3 (Closing-Opening $\varphi\gamma$) where a few more artifacts remain in the image, in this case, in the white part.

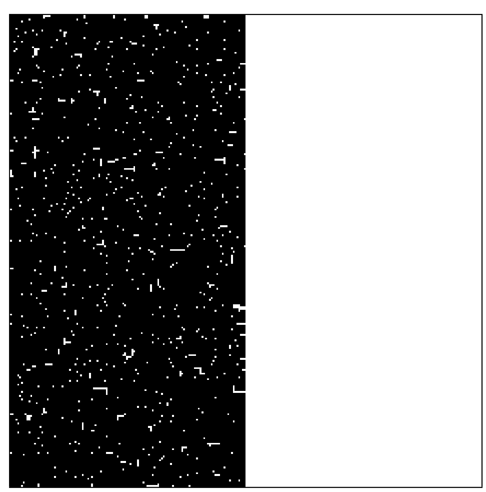
As we can see in Filters 1 (Opening γ) and 2 (Closing φ), each operation removes artifacts from one of the parts, and thus, the combinations of both are the ones that obtain better results.



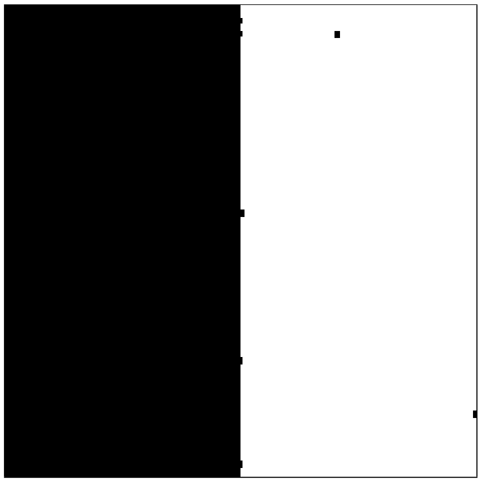
Orignal image: `isn_256.pgm`



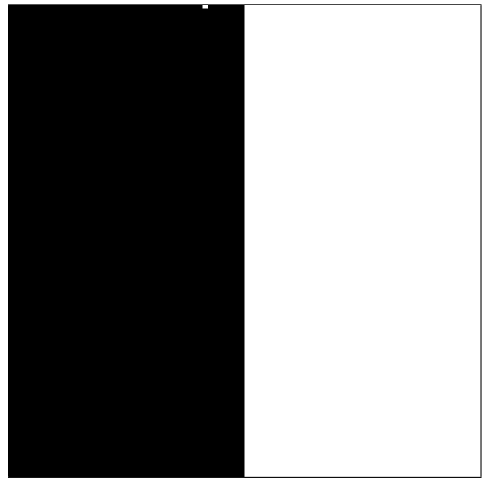
Filter 1: Opening $\gamma_B(I)$



Filter 2: Closing $\varphi_B(I)$



Filter 3: Closing-Opening $\varphi_B\gamma_B(I)$



Filter 4: Opening-Closing $\gamma_B\varphi_B(I)$

Figure 1: Images from Exercise 8a

Exercise 9a

Proof the idempotence of the 'closing-opening' alternated filter.

To prove the idempotence of the closing-opening ($\varphi_B \gamma_B$) we first need to assume some statements:

- Both γ_B (opening) and φ_B (closing) are increasing and idempotent.
- γ_B (opening) is anti-extensive.
- φ_B (closing) is extensive.

Theorem 1. *Idempotence of the 'closing-opening' alternated filter:*

$$\varphi_B \gamma_B \varphi_B \gamma_B(I) = \varphi_B \gamma_B(I)$$

Proof. The proof is done showing that both of the inequalities are satisfied.

$\boxed{\varphi_B \gamma_B \varphi_B \gamma_B(I) \leq \varphi_B \gamma_B(I)}$

$$\begin{aligned} \varphi_B \gamma_B(I) = \varphi_B \gamma_B(I) \Rightarrow \\ \varphi_B \varphi_B \varphi_B \gamma_B(I) = \varphi_B \gamma_B(I) \Rightarrow \\ \varphi_B \gamma_B \varphi_B \gamma_B(I) \leq \varphi_B \gamma_B(I) \end{aligned} \tag{1}$$

$\boxed{\varphi_B \gamma_B \varphi_B \gamma_B(I) \geq \varphi_B \gamma_B(I)}$

$$\begin{aligned} \varphi_B \gamma_B(I) = \varphi_B \gamma_B(I) \Rightarrow \\ \varphi_B \gamma_B \gamma_B \gamma_B(I) = \varphi_B \gamma_B(I) \Rightarrow \\ \varphi_B \gamma_B \varphi_B \gamma_B(I) \geq \varphi_B \gamma_B(I) \end{aligned} \tag{2}$$

By (1) and (2) it can be concluded that the equality $\varphi_B \gamma_B \varphi_B \gamma_B(I) = \varphi_B \gamma_B(I)$ holds. $\square$