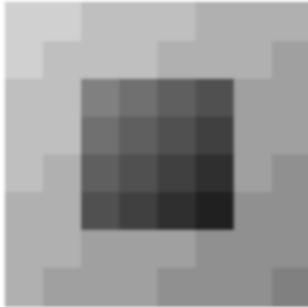


L'image de la figure suivante est une image à niveaux de gris de taille 8×8 pixels et dont les valeurs des niveaux de gris sont codées sur 4 bits. Cette image représente une forme rectangulaire sur un fond.



	0	1	2	3	4	5	6	7
0	13	13	12	12	12	11	11	11
1	13	12	12	12	11	11	11	10
2	12	12	8	7	6	5	10	10
3	12	12	7	6	5	4	10	10
4	12	11	6	5	4	3	10	9
5	11	11	5	4	3	2	9	9
6	11	11	10	10	10	9	9	9
7	11	10	10	10	9	9	9	9

Réponses :

1) Les trois types de transformation sur les images.

- Transformations ponctuelles : par exemple négatif d'une image.
- Transformations locales : par exemple l'opération de convolution.
- Transformations globales : par exemple la segmentation d'image par l'algorithme d'Otsu.

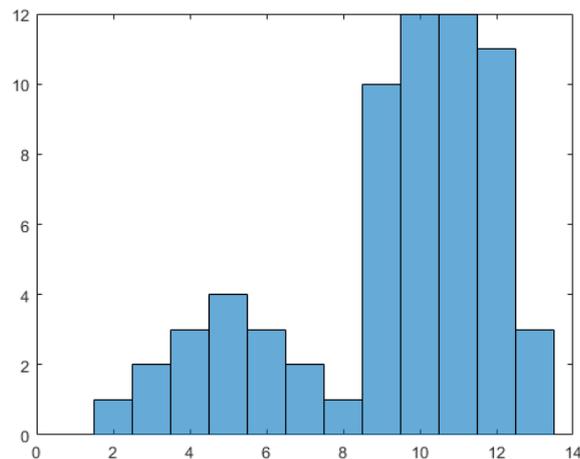
2) Principe de fonctionnement du filtre Gaussien :

Le filtre Gaussien est filtre spatial qui modifie une image en remplaçant la valeur de chaque pixel par une fonction des valeurs de ce pixel ainsi que ses voisins.

Son rôle :

Il est généralement utilisé pour le lissage des images (image smoothing) qui est une opération nécessaire dans plusieurs applications de traitement d'image telles que l'atténuation de bruit, la restauration d'image.

3) Histogramme :



4) Fonction qui permet le changement de la dynamique :

```
Pour i=1 à nblig
    Pour j=1 à nbcoll
        I'(i,j) = 255*(I(i,j)-minI)/(maxI-minI);
```

5) Type des filtres H_1 et H_2 :

- H_1 : filtre passe haut (accentuation ou détection de contours)
- H_2 : filtre passe bas (lissage)

6) Il faut utiliser le filtre H_1 .

Les différentes étapes nécessaires afin de réaliser cette détection :

1. Filtrer l'image d'entrée avec les deux filtres H_1 :

$$g_x = f * S_x$$
$$g_y = f * S_y$$

2. Calculer le gradient en chaque point de l'image :

$$\|\nabla g\| = \sqrt{(g_x)^2 + (g_y)^2}$$

3. Calculer $c(x,y)$ en utilisant un seuil s .

$$c(x,y) = \begin{cases} 1 & \text{si } \|\nabla g\| > s \\ 0 & \text{sinon} \end{cases}$$

7) Les résultats de la convolution du filtre H_1 sur les pixels de l'image I de coordonnées :

(1,1)	-7
(2,2)	-16
(3,3)	-8
(4,4)	-8

(5,5)	17
(6,6)	7
(2,5)	-4
(5,2)	-4

On peut constater qu'il y a beaucoup de bruits. Pour améliorer le résultat, on peut utiliser un filtre moyenneur ou un filtre Gaussien (par exemple le filtre H_2).

8)

Application du filtre H_2 sur le pixel de l'image I de coordonnées (2,2) = 9.6.

Application du filtre médian de taille 3x3 sur le pixel de l'image I de coordonnées (2,2) = 12

On peut dire que le filtre médian n'a pu garder l'information concernant le contour du carré à l'intérieur de l'image. Par contre le filtre H_1 a pu le faire.

9) Seuil choisi $S = 8$.

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	1	1	1	1	0	0
3	0	0	1	1	1	1	0	0
4	0	0	1	1	1	1	0	0
5	0	0	1	1	1	1	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0

10) Soit I' l'image obtenu en binarisant l'image I et soit I'' l'inverse de I' . I'' peut être calculé comme suit :

a)

```
Pour i=1 à nblig
  Pour j=1 à nbcol
    I''(i,j) = 1-I'(i,j);
```

b)

```
Pour i=1 à nblig
  Pour j=1 à nbcol
    I''(i,j) = NOT(I'(i,j));
```

	0	1	2	3	4	5	6	7
0	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
2	1	1	0	0	0	0	1	1
3	1	1	0	0	0	0	1	1
4	1	1	0	0	0	0	1	1
5	1	1	0	0	0	0	1	1
6	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1