

RdF : TD sur les filtres de Laws

1 Filtres de Laws

En 1980, Kenneth Laws a été le premier chercheur à proposer de calculer des indices de texture en utilisant une technique de filtrage linéaire par convolution. Par la suite, les filtres de Laws ont servi de référence pour la comparaison des performances d'autres indices de texture obtenus par filtrage, par exemple avec des filtres de Gabor ou des filtres utilisant d'autres ondelettes.

La technique décrite par Laws dans son article de 1980 permet de traiter des images au moyen de filtres de convolution en deux dimensions. Dans cet exercice, nous traiterons le cas de textures mono-dimensionnelles et de filtres également mono-dimensionnels. Le principe est identique et l'extension en 2D est très simple. Les 5 filtres mono-dimensionnels de Laws sont définis par les masques suivants :

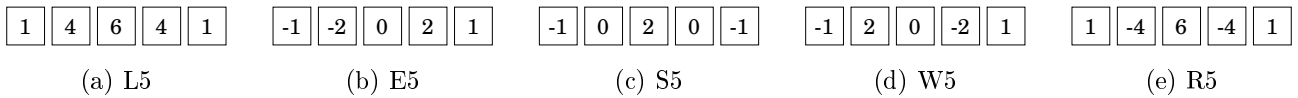


FIGURE 1 – masques des filtres de Laws 1D

Les images initiales sont binaires, mais le résultat des opérations de convolution est un nombre entier relatif. Afin de conserver une symétrie dans les valeurs, on considèrera que les deux états binaires sont codés par -1 pour le noir (au lieu de 0) et $+1$ pour le blanc.

1. calculer les 5 images résultant du filtrage d'une image ligne ne contenant que des pixels noirs. Même question pour une ligne ne contenant que des pixels blancs.
2. calculer les 5 images résultant du filtrage d'une image ligne contenant une alternance de pixels noirs et blancs (série $[-1, +1, -1, +1, -1, \text{etc.}]$).
3. combien d'alternances $+/-$ trouve-t-on dans la série de coefficients définissant chaque filtre ?
4. pour chaque filtre de Laws, peut-on trouver une texture, c'est-à-dire un arrangement périodique de valeurs binaires, pour lequel la valeur absolue de la réponse est *le plus souvent* maximale ?

2 Indices de texture

On ne peut parler de texture que si les alternances noir/blanc sont régulières et s'étendent sur plus d'un ou deux pixels. De ce fait, Laws propose de calculer des indices de texture en sommant les valeurs absolues des sorties de filtres sur une *fenêtre* de quelques pixels de largeur. Dans son article, il propose d'utiliser une fenêtre de 15×15 pixels pour des textures 2D. Ici, comme les calculs sont réalisés manuellement, nous considérerons des fenêtres 1D de taille 4.

Si $R(x)$ désigne la sortie d'un filtre pour le pixel situé à la position x , l'indice de texture $T(x)$ pour ce filtre et ce pixel est calculé par :

$$T(x) = \sum_{i=-1}^{+2} |R(x+i)| . \quad (1)$$

1. pour chaque filtre, calculer le coefficient de pondération qui permet de garantir que l'indice de texture prend toujours une valeur comprise entre 0 et 1 ;
2. calculer les indices de texture obtenus pour la ligne de pixels de la figure 2 pour le filtre *L5* normalisé ; pour le filtre *R5* normalisé ; conclure.

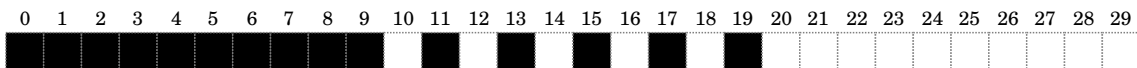


FIGURE 2 – ligne d'une image binaire