

Chapitre II

Prétraitements

Prétraitements

- Transformations géométriques
 - Affines
 - Polynomiales
- – Filtrage
 - Linéaire : convolution
 - Non linéaire : Médian
 - Morphologique
- – Binarisation
 - Seuillage global
 - Seuillage local

Transformations géométriques

But

- Réaliser une correction des distorsions provenant des artefacts de l'acquisition
 - Opérations image à image

Formulation

- Les transformations géométriques peuvent être formulées de manière très générale.
- Soit $I(x,y)$ l'image d'origine et $I'(x',y')$ sa version distordue. Les deux images sont liées par les équations
 - $x' = f_1(x,y); y' = f_2(x,y)$
- Le type de distorsion est caractérisé par le type de $f_1(.,.)$, $f_2(.,.)$
 - Si on connaît f_1 et f_2 , on peut corriger l'image

Transformations géométriques

- Les transformations affines (linéaires)
 - Couvrent la translation, la mise à l'échelle (scaling), la rotation et l'inclinaison
 - Peuvent être représentées par la notation matricielle

la connaissance des a_{ij} peut permettre la correction

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

Transformations géométriques

La tableau suivant résume les propriétés des transformations affines :

Type	Propriétés	Signification
Translation	$a_{ii} = 1, i=1,2$ $a_{ij} = 0, i \neq j$	
Mise à l'échelle	$a_{12} = a_{21} = 0$	
Rotation	$a_{11} = \cos \alpha, a_{12} = -\sin \alpha$ $a_{21} = \sin \alpha, a_{22} = \cos \alpha$	α : angle de rotation
Inclinaison-Biais	$A_{11} = 1, a_{12} = \tan \beta$ $A_{21} = 0, a_{22} = 1$	β : angle d'inclinaison

Correction de l'inclinaison d'un document

Projection d'histogramme

Principe:

- Comme on ne connaît pas l'angle d'inclinaison (θ), on projette l'image sur différents axes

Correction de l'inclinaison d'un document

Projection d'histogramme

- Exemple

يهدف البحث إلى دراسة الخواص الحرارية
والصوتية والفيكتيكية لعادة البوليفر والبحث
عن تغير خواصها بفعل العوامل المؤثرة تكيّف
نعرف مدى استجابتها للمؤثرات الخارجية

يهدف البحث إلى دراسة الخواص الحرارية
والصوتية والفيكتيكية لعادة البوليفر والبحث
عن تغير خواصها بفعل العوامل المؤثرة تكيّف
نعرف مدى استجابتها للمؤثرات الخارجية

Correction de l'inclinaison d'un document

Projection d'histogramme

- L'histogramme H pour un angle de projection α est :

$$H(yl, \alpha) = \sum_0^N i(xk, yl)$$

$$H(yl, \alpha) = \sum_0^N i'(\cos \alpha \cdot xk - \sin \alpha \cdot yl, \sin \alpha \cdot xk + \cos \alpha \cdot yl)$$

- avec $i'(x'k, y'l) = 0$ si $(x'k, y'l)$ est en dehors du document, et $i'(x'k, y'l) = I(\text{round}(x'), \text{round}(y'))$ autrement.

Correction de l'inclinaison d'un document

Projection d'histogramme

- La variation d'histogramme entre les lignes consécutives le long de la direction spécifiée par (α) est comme suit :

$$A(\alpha) = \sum_0^N [H(y_l, \alpha) - H(y_{l+1}, \alpha)]^2$$

Correction de l'inclinaison d'un document

Projection d'histogramme

- La variation d'histogramme entre les lignes consécutives le long de la direction spécifiée par (α) est comme suit :

$$A(\alpha) = \sum_0^N [H(y_l, \alpha) - H(y_{l+1}, \alpha)]^2$$

Correction de l'inclinaison d'un document

Projection d'histogramme

- La variation d'histogramme entre les lignes consécutives le long de la direction spécifiée par (α) est comme suit :

$$A(\alpha) = \sum_0^N [H(y_l, \alpha) - H(y_{l+1}, \alpha)]^2$$

L'angle d'inclinaison estimé est donné par : $\alpha^* = \arg \max A(\alpha)$

Réduction du bruit

Origine

- Appareil d'écriture ou instrument d'écriture

Problèmes

- Coupures de segments,
- Ouvertures des boucles,
- Connexions de traits

Solution

- Filtrage

Filtrage d'image

Définition

- Le filtrage d'image a pour but d'améliorer sa qualité
- Chaque filtre cherche à atténuer un type de défaut bien précis
 - > Il n'y a pas de filtre universel capable de corriger tous les défauts
 - > Il faut choisir les bons filtres suivant les défauts que l'on désire corriger

Filtrage d'image

Deux types de filtrage

- Filtrage global
- Filtrage local

Filtrage d'image

- Filtrage global
 - Chaque pixel de la nouvelle image est calculé en prenant en compte la totalité des pixels de l'image de départ
 - Exemple, les opérations sur les histogrammes ou les opérations qui nécessitent de passer dans l'espace de Fourier

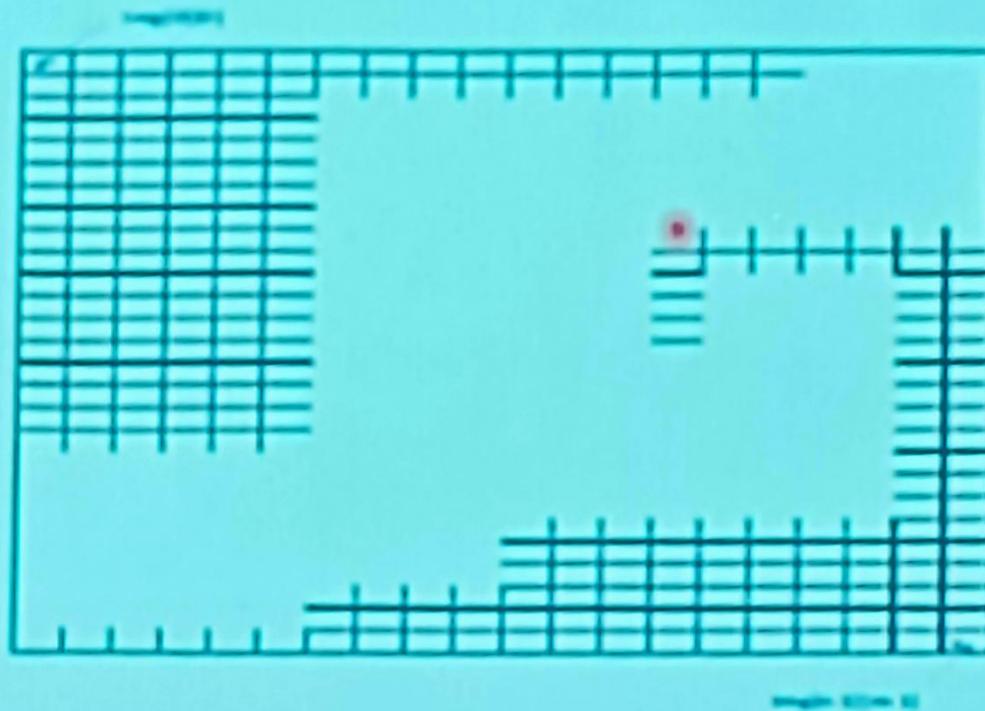
Filtrage d'image

- Filtrage local

- Chaque pixel de la nouvelle image est calculé en prenant en compte seulement un voisinage du pixel correspondant dans l'image d'origine
- Il est d'usage de choisir un voisinage carré et symétrique autour du pixel considéré
- Ces voisinages sont donc assimilables à des tableaux à deux dimensions (matrices) de taille impaire

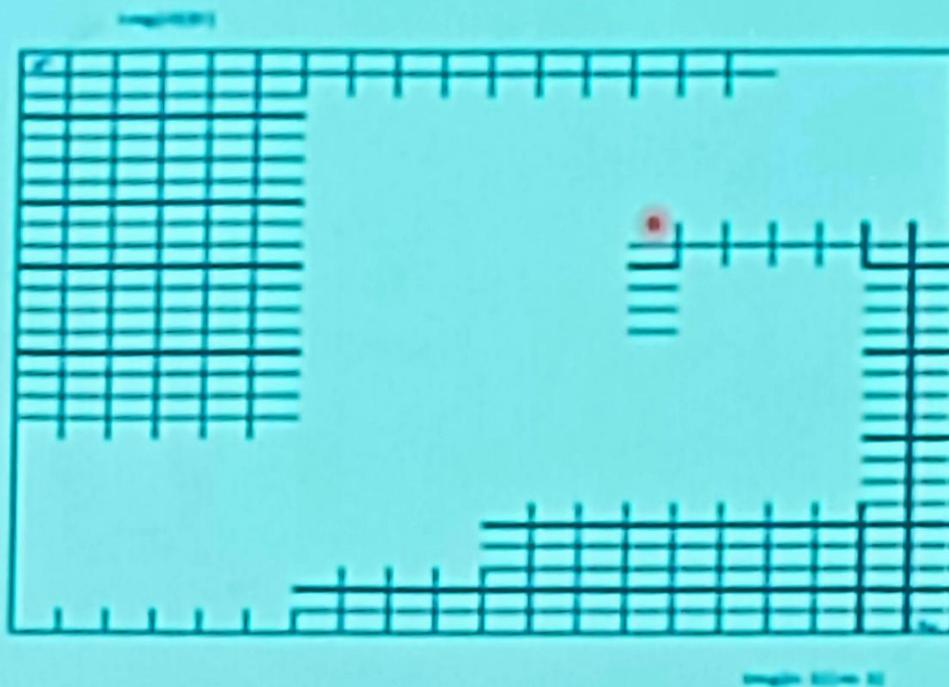
Filtrage Local

Convention de notation



Filtrage Local

Convention de notation



Filtrage Local

- Voisinage 3×3

$P_{x-1,y-1}$	$P_{x,y-1}$	$P_{x+1,y-1}$
$P_{x-1,y}$	$P_{x,y}$	$P_{x+1,y}$
$P_{x-1,y+1}$	$P_{x,y+1}$	$P_{x+1,y+1}$

Filtrage Local

- Voisinage 5*5

$P_{x-2,y-2}$	$P_{x-1,y-2}$	$P_{x,y-2}$	$P_{x+1,y-2}$	$P_{x+2,y-2}$
$P_{x-2,y-1}$	$P_{x-1,y-1}$	$P_{x,y-1}$	$P_{x+1,y-1}$	$P_{x+2,y-1}$
$P_{x-2,y}$	$P_{x-1,y}$	$P_{x,y}$	$P_{x+1,y}$	$P_{x+2,y}$
$P_{x-2,y+1}$	$P_{x-1,y+1}$	$P_{x,y+1}$	$P_{x+1,y+1}$	$P_{x+2,y+1}$
$P_{x-2,y+2}$	$P_{x-1,y+2}$	$P_{x,y+2}$	$P_{x+1,y+2}$	$P_{x+2,y+2}$

Filtrage Local

Filtrage local linéaire

- Le filtre local est dit linéaire si la valeur du nouveau pixel est une combinaison linéaire des valeurs des pixels du voisinage

Filtrage Local

Filtrage local linéaire

- Le filtre local est dit linéaire si la valeur du nouveau pixel est une combinaison linéaire des valeurs des pixels du voisinage

$$Nv_{x,y} = \sum_{i,j} A_{i,j} * P_{x+i,y+j}$$

- Combinaison linéaire des pixels du voisinage, avec i,j variant entre -k et +k,
- k=1 pour un voisinage de 3x3, k=2 pour 5x5 etc.
- A_{i,j} = valeur entière ou réelle, coefficients spécifiques au filtre linéaire

Filtrage Local

Normalisation

- Si la valeur obtenue n'est pas dans les limites imposées par le format d'image (entier(s) entre 0 et 255), alors la valeur doit être normalisée

$$FN = \left| \sum_{i,j} A_{i,j} \right|$$

Filtrage Local

Noyau

- Il est d'usage de présenter les coefficients sous forme d'une matrice (appelée noyau de convolution) facilitant ainsi la mise en correspondance avec les valeurs du voisinage

A _{-1,-1}	A _{0,-1}	A _{1,-1}
A _{-1,0}	A _{0,0}	A _{1,0}
A _{-1,+1}	A _{0,1}	A _{1,1}

Kernel 3*3

P _{x-1,y-1}	P _{x,y-1}	P _{x+1,y-1}
P _{x-1,y}	P _{x,y}	P _{x+1,y}
P _{x-1,+1}	P _{x,y+1}	P _{x+1,y+1}

Voisinage 3*3

Filtrage Local

Noyau*Voisinage (multiplication point à point)

$A_{-1,-1} * P_{x-1,y-1}$	$A_{0,-1} * P_{x,y-1}$	$A_{1,-1} * P_{x+1,y-1}$
$A_{-1,0} * P_{x-1,y}$	$A_{0,0} * P_{x,y}$	$A_{1,0} * P_{x+1,y}$
$A_{-1,+1} * P_{x-1,y+1}$	$A_{0,1} * P_{x,y+1}$	$A_{1,1} * P_{x+1,y+1}$

Filtrage Local

Noyau*Voisinage (multiplication point à point)

$A_{-1,-1} * P_{x-1,y-1}$	$A_{0,-1} * P_{x,y-1}$	$A_{1,-1} * P_{x+1,y-1}$
$A_{-1,0} * P_{x-1,y}$	$A_{0,0} * P_{x,y}$	$A_{1,0} * P_{x+1,y}$
$A_{-1,+1} * P_{x-1,y+1}$	$A_{0,1} * P_{x,y+1}$	$A_{1,1} * P_{x+1,y+1}$

La nouvelle valeur du pixel $P_{x,y}$ est alors la somme des éléments de la matrice / FN

Filtrage Local

- Exemple:



$$P = (0 * 2 + 0 * 0 + 0 * 0) + (0 * 0 + 1 * 0 + 3 * 0) + (0 * 0 + 2 * 0 + 1 * (-2))$$

$$P = -2$$

Filtrage Local

Exemple:

- Flou uniforme
- – Description : moyenne arithmétique des valeurs du voisinage

1	1	1
1	1	1
1	1	1

Filtrage Local

Exemple:

- Flou gaussien
 - Description : Atténuation des changements brusques d'intensité
 - Principe : Les pixels du voisinage qui sont proches du pixel central ont un poids plus fort (= plus d'influence) que ceux qui sont plus éloignés

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

Filtrage Local

PREWITT

- – Description : Détection des contours
- – Noyau : les coefficients symétriques par rapport à l'axe vertical/vertical sont de signes opposés. La somme des coefficients est nulle

Filtrage Local

PREWITT

-1	0	1
-1	0	1
-1	0	1

Détection contours verticaux

-1	-1	-1
0	0	0
1	1	1

Détection contours horizontaux

Filtrage Local

Sobel

Detection de contours

-1	0	1
2	0	2
-1	0	1

-1	2	-1
0	0	0
1	2	1

Détection contours verticaux

Détection contours horizontaux

Filtrage Local

Le filtre de Canny

Détecteur de Canny: détection des contours

Avantages:

1. *Bonne détection* : faible taux d'erreur dans la signalisation des contours,
2. *Bonne localisation* : minimisation des distances entre les contours détectés et les contours réels,
3. *Clarté de la réponse* : une seule réponse par contour et pas de faux positifs

Filtrage Local

Canny

Mise en oeuvre

1. Réduction du bruit

Un filtrage gaussien 2D est utilisé :

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2

Filtrage Local

Canny

Mise en oeuvre

2. Gradient d'intensité

Deux masques de convolution, un de dimension 3×1 et l'autre 1×3 :

$$G_x = [-1 \ 0 \ 1] \quad G_y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

Filtrage Local

Canny

Mise en oeuvre

2. Gradient d'intensité

Deux masques de convolution, un de dimension 3×1 et l'autre 1×3 :

$$G_x = [-1 \ 0 \ 1] \quad G_y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

La valeur du gradient en un point est:

$$|G| = \sqrt{G_x^2 + G_y^2}$$

Filtrage Local

Canny

Mise en oeuvre

3. Direction des contours

Les orientations des contours sont déterminées par la formule :

Filtrage Local

Canny

Mise en oeuvre

3. Direction des contours

Les orientations des contours sont déterminées par la formule :

$$\theta = \pm \arctan \left(\frac{G_x}{G_y} \right)$$

Filtrage local non linéaire

Filtre Médian

- Le filtre médian remplace un pixel par la valeur médiane des pixels avoisinants.
- $V_T = \text{tri dans l'ordre croissant des Pixels dans le voisinage avec } P \text{ inclus}$
 $P = V_T(5)$

Filtrage local non linéaire

Filtre Médian

- Exemple:

6	7	24
4	9	12
6	45	3



6	7	24
4	7	12
6	45	3

- $VT = (3, 4, 6, 6, 7, 9, 12, 24, 45)$

Filtrage local non linéaire

Filtre Majorité

- Nettoyage
- Cette technique est très importante car parfois malgré les multiples filtres appliqués on se retrouve avec des pixels isolés, donc un nettoyage s'impose pour supprimer ces pixels.
- Majorité : Cette technique consiste à modifier la valeur d'un pixel blanc(1) en un pixel noir(0) si son voisinage 3X3 contient au moins cinq pixels noirs(0) et vis-versa.

Transformations Morphologiques

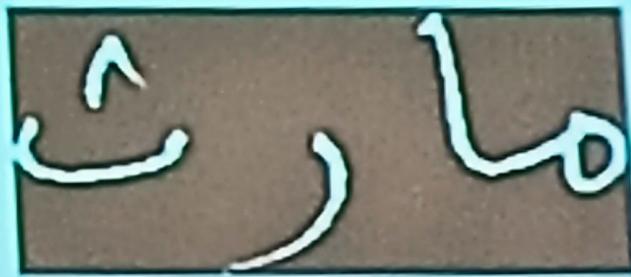
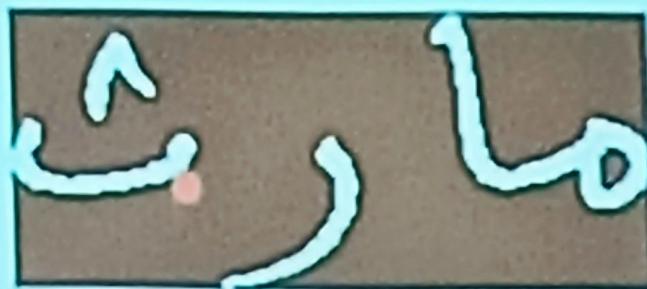
- Les transformations morphologiques sont de simples opérations basées sur la forme de l'image.
- Applicable normalement sur des images binaires.
- Deux entrées sont nécessaires :
 - Image originale,
 - structuring element or kernel qui decide de la nature de l'operation.
- Deux opérateurs morphologiques de base sont l'érosion et la dilatation.

Erosion

- Inverser l'image pour avoir le premier plan (écriture) en blanc.
- Le noyau glisse à travers l'image.
- Un pixel de l'image d'origine (soit 1 soit 0) ne sera considéré comme 1 que si tous les pixels sous le noyau sont 1, sinon il est érodé (mis à zéro).
- Tous les pixels proches de la limite seront supprimés en fonction de la taille du noyau. Ainsi, l'épaisseur ou la taille de l'objet diminue ou simplement la région blanche diminue dans l'image (image inverse).

Erosion

- Exemple:



Dilatation

- C'est l'opposé de l'érosion.
- Un pixel vaut "1" si au moins un pixel sous le noyau vaut "1".
- Ainsi, la zone blanche de l'image ou la taille de l'objet de premier plan augmente.

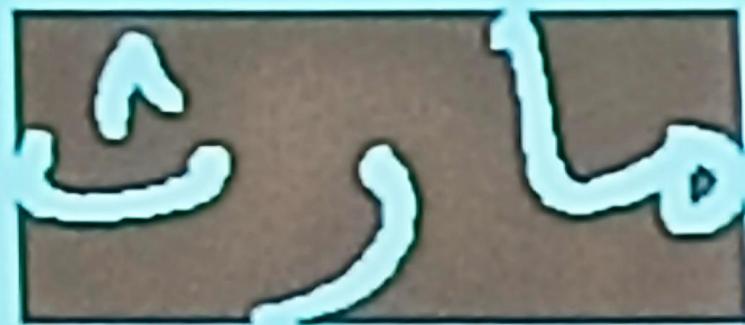
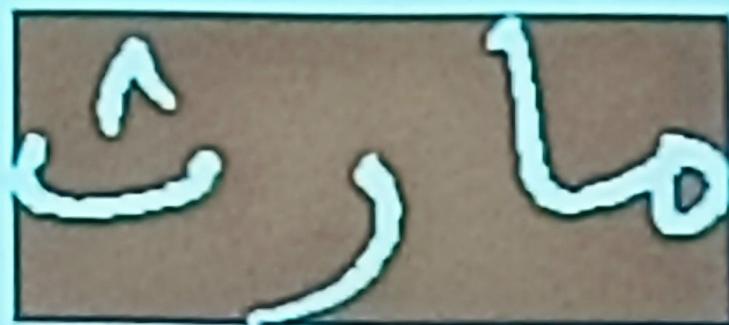
Dilatation

Remarque

- Dans des cas comme l'élimination du bruit, l'érosion est suivie d'une dilatation.
- L'érosion supprime les bruits blancs, mais elle rétrécit aussi notre objet.
- Alors on le dilate. Puisque le bruit a disparu, il ne reviendra pas, mais notre zone d'objet augmente.
- La dilatation est également utile pour joindre des parties brisées d'un objet.

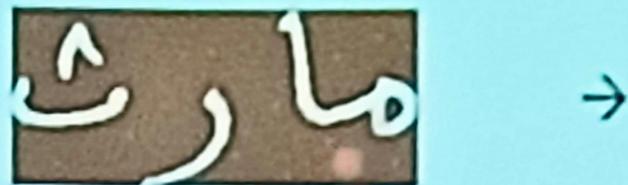
Dilatation

- Exemple:



Ouverture

- Erosion suivie d'une dilatation
- Eliminer les bruits
- Exemple:



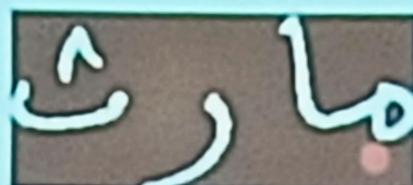
Fermeture

- Dilatation suivie d'une Erosion
- Eliminer les trous occasionnels dans une images
- Exemple:

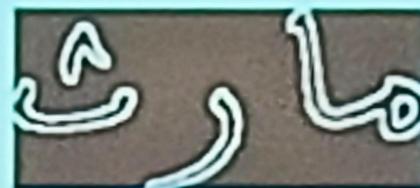


Gradient morphologique

- La différence entre la dilatation et l'érosion d'une image
- Le résultat ressemblera à un contoure mais pas nécessairement de dimension 1 pixel.
- Exemple:



→



Succession de filtres

- Il s'agit de choisir la succession de filtres par lesquels on fait passer notre image pour un meilleur rendu et de meilleurs résultats dans les traitements ultérieurs.

Structuring Element ou Kernel

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

0	0	0	1	0	0	0
0	0	0	1	0	0	0
0	0	0	1	0	0	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	0	0	1	0	0	0
0	0	0	1	0	0	0

0	0	0	1	0	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	0	1	0	0	0

Binarisation

Objectif:

- Détacher la forme (le texte) du fond qui n'est pas utile à la reconnaissance
- La plupart des méthodes existantes travaillent sur une image simplifiée noire et blanche

Binarisation

Binarisation par seuillage

- C'est la méthode la plus simple et la plus utilisée
- Il y a une relation entre les niveaux de gris d'un pixel et son appartenance ou non à une forme
- Mais cette relation n'est pas toujours évidente

Binarisation

Mise en œuvre

- Les méthodes de binarisation consistent à transformer une image de niveaux de gris en une image noire et blanche.
- Soit f la fonction de binarisation et $i(x,y)$ un pixel de l'image, on a :

$$f(i(x,y)) \in \{0,1\}$$

- Les méthodes de binarisation se divisent en deux classes : globales et locales.

Binarisation

Mise en œuvre

- Les méthodes globales
 - calculent un seul seuil pour toute l'image. Les pixels ayant un niveau de gris plus foncé que le seuil sont mis à noir et les autres à blanc
- Les méthodes locales
 - calculent un seuil pour chaque pixel en fonction de l'information contenue dans son voisinage

Binarisation

Méthodes globales

- Les méthodes de seuillage global utilisent une seule valeur de seuil d'intensité.
- Cette valeur est calculée sur la base d'heuristiques ou de statistiques d'attributs d'image globaux afin de classer les pixels de l'image en pixels de premier plan (texte) ou d'arrière-plan (non textuel)

Binarisation

Méthodes globales

Méthode de OTSU

- Otsu formule le problème comme une analyse discriminante
- Il utilise une fonction critère particulière comme mesure de séparation statistique
- Il fait l'hypothèse qu'il y a 2 classes dans l'histogramme
- Des statistiques sont calculées pour les deux classes de valeurs d'intensité (fond et avant-plan) séparées par un seuil intensité
- On calcule les statistiques pour chaque niveau d'intensité i , c.à.d. pour tous les seuils possibles. Le niveau qui minimise la fonction critère est choisi comme seuil

Binarisation

Méthodes globales

- La méthode d'Otsu
- Soit 2 classes de pixels C_1 et C_2 . La classe C_1 est définie comme étant composée des pixels ayant une valeur comprise entre 0 et k avec $k < 255$.
- Le reste (classe C_2) des pixels compris entre k et 255 faisant partie de la seconde classe. Le but de la méthode est de trouver " k " tel qu'il sépare au mieux le fond et les objets de l'image traitée.

Binarisation

Méthodes globales

La méthode d'Otsu

- Afin de trouver "k", la méthode d'Otsu va calculer la variance inter-classe entre C1 et C2 pour tous les k possibles (de 0 à 255).
- La variance inter-classe caractérise la dissemblance qui existe entre les pixels des deux classes.
- Plus elle est haute, moins les deux classes se ressemblent. Et vice versa.

Binarisation

Méthodes globales

La méthode d'Otsu

- Par conséquent, le seuil optimal est le k qui permet d'obtenir la plus haute variance inter-classe.

La méthode d'Otsu

- Les étapes de l'algorithme d'Otsu sont les suivantes :
 1. Construction de l'histogramme des niveaux de gris de l'image.
 2. Normalisation de l'histogramme obtenu en 1. Normaliser l'histogramme, permet d'obtenir un histogramme dont toutes les valeurs sont comprises entre 0 et 1. Le calcul ci-dessous permet d'y parvenir :

$$\text{hist} = (\text{hist} - \min) / (\max - \min)$$

La méthode d'Otsu

- Remarque : en normalisant l'histogramme, on obtient une distribution de probabilité.
- Par exemple la probabilité que dans notre image il y ait un pixel dont le niveau de gris est 255 est la valeur de notre histogramme normalisé pour l'abscisse 255.

La méthode d'Otsu

3. Calcul de la probabilité qu'un pixel de l'image appartienne à C1. Cela est fait en calculant la somme de toutes les colonnes de l'histogramme comprises entre 0 et k.
4. Calcul de la probabilité qu'un pixel de l'image appartienne à C2. En faisant simplement $1 - \text{proba}(C1)$ calculé dans l'étape précédente.

La méthode d'Otsu

5. Calcul de la variance inter-classe entre C1 et C2 :

$$var_{c1 \text{ inter } c2} = \frac{(Moy * Proba_{c1}(k) - Moy_{c1}(k))^2}{Proba_{c1}(k) * Proba_{c2}(k)}$$

avec Moy étant la moyenne de l'image, $Proba_{C1}(k)$ étant la probabilité qu'un pixel de l'image appartienne à C1, $Moy_{c1}(k)$ la moyenne des pixels de la classe C1.

La méthode d'Otsu

6. Les étapes 2, 3, 4, 5 sont répétées pour toutes les valeurs de k possibles.

Binarisation

Méthodes locales

Niblack et Sauvola

- Niblack et Sauvola thresholds sont des méthodes de seuillage locales utiles pour les images où le background n'est pas uniforme, spécialement pour la reconnaissance de textes.
- Au lieu de calculer un seul seuil global pour l'image entière, plusieurs seuils sont calculés pour chaque pixel en utilisant une formule spécifique qui tient compte de la Moyenne et de l'écart-type du voisinage local centeré autour du pixel.

Binarisation Méthodes locales

Niblack

Le seuil calculé pour le pixel (x,y) est :

$$T(x,y) = m(x,y) + k\sigma(x,y)$$

où $m(x,y)$ et $\sigma(x,y)$ sont respectivement la moyenne et l'écart-type calculés dans un voisinage local de (x,y)

Binarisation

Méthodes locales

Niblack

- La taille du voisinage doit être suffisamment petite pour préserver les détails locaux, mais suffisamment large pour supprimer le bruit
- La valeur de k est utilisée pour ajuster la partie de l'objet «print» totale considérée comme appartenant à un objet donné

Binarisation

Méthodes locales

Niblack

Si on note $f(x,y)$ le niveau de gris dans un point (x,y) , alors l'écart type local $\sigma(x,y)$ dans un voisinage de taille $(2k_1+1) \times (2k_2+1)$ autour de (x,y) peut être calculé comme :

$$\sigma^2(x,y) = \frac{1}{(2k_1+1)(2k_2+1)} \sum_{m=-k_1}^{k_1} \sum_{n=-k_2}^{k_2} (f(x+m, y+n))^2 - (m(x,y))^2$$

Binarisation Méthodes locales

Niblack

Le seuil calculé pour le pixel (x,y) est :

$$T(x, y) = m(x, y) + k\sigma(x, y)$$

où $m(x, y)$ et $\sigma(x, y)$ sont respectivement la moyenne et l'écart-type calculés dans un voisinage local de (x, y)

Binarisation

Méthodes locales

Niblack

Si on note $f(x,y)$ le niveau de gris dans un point (x,y) , alors l'écart type local $\sigma(x,y)$ dans un voisinage de taille $(2k_1+1) \times (2k_2+1)$ autour de (x,y) peut être calculé comme :

$$\sigma^2(x,y) = \frac{1}{(2k_1+1)(2k_2+1)} \sum_{m=-k_1}^{k_1} \sum_{n=-k_2}^{k_2} (f(x+m, y+n))^2 - (m(x,y))^2$$

*

Binarisation

Méthodes locales

Niblack

Si on note $f(x,y)$ le niveau de gris dans un point (x,y) , alors l'écart type local $\sigma(x,y)$ dans un voisinage de taille $(2k_1+1) \times (2k_2+1)$ autour de (x,y) peut être calculé comme :

$$\sigma^2(x,y) = \frac{1}{(2k_1+1)(2k_2+1)} \sum_{m=-k_1}^{k_1} \sum_{n=-k_2}^{k_2} (f(x+m, y+n))^2 - (m(x,y))^2$$

- où $m(x,y)$ est la valeur moyenne de $f(x,y)$ dans le voisinage

$$m(x,y) = \frac{1}{(2k_1+1)(2k_2+1)} \sum_{m=-k_1}^{k_1} \sum_{n=-k_2}^{k_2} f(x+m, y+n)$$

Binarisation

Méthodes locales

Méthode de Sauvola

- Niblack ne fonctionne pas sur un fond clair
- les valeurs des éléments non désirés peuvent dépasser les seuils. De plus, l'étape de post-traitement est coûteuse
- Modification : tenir compte de la variance locale (SAUVOLA)
- Seuil proposé : $T(x,y) = m(x,y) * [1 + k * \sigma(x,y)/R]$ où R est la variance

Binarisation

Méthodes locales

Méthode de Sauvola

- Niblack ne fonctionne pas sur un fond clair
- les valeurs des éléments non désirés peuvent dépasser les seuils. De plus, l'étape de post-traitement est coûteuse
- Modification : tenir compte de la variance locale (SAUVOLA)
- Seuil proposé : $T(x,y) = m(x,y) * [1 + k * \sigma(x,y)/R]$ où R est la variance
- La multiplication des deux termes par la moyenne locale a pour effet d'amplifier la contribution de la variance de manière adaptative

Binarisation

Méthodes locales

Méthode de Sauvola

- Si on considère par exemple un texte foncé sur un fond clair, mais avec du bruit, m fait décroître la valeur du seuil dans les régions du fond
- L'effet de cette méthode est d'effacer d'une manière efficace le bruit dans une image seuillée