D'abord, les types d'mages les plus utilisés sont:   
-image en noir et blanc (gray)   
-image RGB (rgb)   
-image indexée (ind)   
Voici les transformations possibles et les fonctions correspondantes à utiliser:   
-image rgb vers gray: fonction: rgb2gray   
-image rgb vers ind: rgb2ind   
-image gray vers rgb: gray2rgb   
-image gray vers ind: gray2ind   
-image ind vers gray: ind2gray   
-image ind vers rgb: ind2rgb

Imhist : histogramme

histeq :Égalisation d’histogramme

L = bwlabel (BW, n) renvoie une matrice L, de même taille que BW, contenant des étiquettes pour les objets connectés en BW. La variable n peut avoir une valeur de 4 ou 8, où 4 spécifie 4 objets connectés et 8 spécifie 8 objets connectés. Si l'argument est omis, il est par défaut 8.

Les éléments de L sont des valeurs entières supérieures ou égales à 0. Les pixels marqués 0 sont l'arrière-plan. Les pixels étiquetés 1 forment un objet; Les pixels étiquetés 2 constituent un second objet; etc.  
J = adapthisteq (I) améliore le contraste de l'image en niveaux de gris I en transformant les valeurs en utilisant une égalisation d'histogramme adaptatif à contraste limité

im2bw

BW = im2bw (I, level) convertit l'image en niveaux de gris I en une image binaire. L'image de sortie BW remplace tous les pixels de l'image d'entrée par une luminance supérieure au niveau par la valeur 1 (blanc) et remplace tous les autres pixels par la valeur 0 (noire). Spécifiez un niveau dans la plage [0,1]. Cette plage est relative aux niveaux de signal possibles pour la classe de l'image. Par conséquent, une valeur de niveau de 0,5 est à mi-chemin entre le noir et le blanc, indépendamment de la classe. Pour calculer l'argument de niveau, vous pouvez utiliser la fonction graythresh. Si vous ne spécifiez pas de niveau, im2bw utilise la valeur 0.5.

Si l'image d'entrée n'est pas une image en niveaux de gris, im2bw convertit l'image d'entrée en niveaux de gris, puis convertit cette image en niveaux de gris en binaire par seuillage.

Level = graythresh (I) calcule un seuil global (niveau) qui peut être utilisé pour convertir une image d'intensité en une image binaire avec im2bw. Niveau est une valeur d'intensité normalisée qui se trouve dans la plage [0, 1].

bw=edge(BW) : détecter les contours

C = uint8 (a) renvoie la valeur entière mémorisée de l'objet fi a en tant que uint8 intégré. Si la longueur de mot entier stockée est trop grande pour un uint8, ou si l'entier stocké est signé, la valeur renvoyée est saturée à uint8. L'entier stocké est le nombre binaire brut, dans lequel le point binaire est supposé être à l'extrême droite du mot

Imcontour (I) trace un tracé de contour de l'image en niveaux de gris I, en configurant automatiquement les axes pour que leur orientation et leur rapport d'aspect correspondent à l'image.

Imcontour (I, n) trace un tracé de contour de l'image en niveaux de gris I, en configurant automatiquement les axes de sorte que leur orientation et leur rapport d'aspect correspondent à l'image. N est le nombre de niveaux de contour uniformément espacés dans la courbe; Si vous omettez l'argument, le nombre de niveaux et les valeurs des niveaux sont automatiquement sélectionnés.

[C, handle] = imcontour (...) renvoie la matrice de contour C et une poignée à un objet hggroup.

SE = strel (forme, paramètres) crée un élément structurant, SE, du type spécifié par la forme. Ce tableau répertorie toutes les formes prises en charge. Selon la forme, strel peut prendre des paramètres supplémentaires. Reportez-vous aux descriptions de syntaxe qui suivent pour plus de détails sur la création de chaque type d'élément structurant.

SE = strel ('carré', W) crée un élément structurant carré dont la largeur est W pixels. W doit être un entier non négatif scalaire

C = makecform (type) crée la structure de transformation des couleurs C qui définit la conversion de l'espace colorimétrique spécifiée par type. Pour effectuer la transformation, passez la structure de transformation des couleurs en tant qu'argument à la fonction applycform.

L'argument type spécifie l'une des conversions répertoriées dans le tableau suivant. Makecform supporte les conversions entre les membres de la famille des espaces de couleur indépendants de l'appareil définis par la Commission Internationale de l'Éclairage (CIE). De plus, makecform prend également en charge les conversions vers et à partir des espaces de couleur sRGB et CMJN.

B = applycform (A, C) convertit les valeurs de couleur en A en l'espace de couleur spécifié dans la structure de transformation de couleur C. La structure de transformation de couleur spécifie divers paramètres de la transformation. Voir makecform pour plus de détails

Si A est bidimensionnel, chaque ligne est interprétée comme une couleur, sauf si la structure de transformation des couleurs contient un profil ICC en niveaux de gris. (Voir la remarque pour ce cas.) A peut avoir une ou plusieurs colonnes, selon l'espace de couleur d'entrée. B a le même nombre de lignes et 1 ou plusieurs colonnes, selon l'espace de couleur de sortie. (La spécification ICC prend actuellement en charge les espaces de périphérique jusqu'à 15 canaux.)

**bwulterode:**

Euclidienne transformée de distance du complément de BW. La connectivité par défaut pour calculer les maxima régionaux est 8 pour deux dimensions,26 pour trois dimensions, et conndef (ndims (BW), 'maximal') pour les dimensions supérieures.

BW2 = bwulterode (BW, method, conn) spécifie la méthode de transformation de distance et la connectivité maxima régionale. Méthode peut être l'une des chaînes «euclidienne», «cityblock», «échiquier» et «quasi-euclidienne

Double (x) renvoie la valeur de double précision pour X. Si X est déjà un tableau à double précision, le double n'a aucun effet

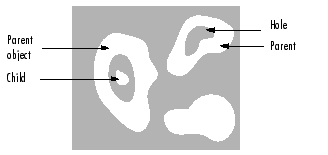
Filter2 : filtrage bidimensionnel

medfilt2 : filtre median

wiener2 : filtre Wiener

**Bwboundaries**

B = bwboundaries (BW) trace les frontières extérieures des objets, ainsi que les limites des trous à l'intérieur de ces objets, dans l'image binaire BW. Bwboundaries descend également dans les objets les plus à l'extérieur (les parents) et trace leurs enfants (objets complètement enfermés par les parents). BW doit être une image binaire où des pixels non nuls appartiennent à un objet et 0 pixels constituent l'arrière-plan. La figure suivante illustre ces composants.



Bwboundaries retourne B, un tableau de cellules P-par-1, où P est le nombre d'objets et de trous. Chaque cellule dans le tableau de cellules contient une matrice Q-by-2. Chaque ligne dans la matrice contient les coordonnées de ligne et de colonne d'un pixel limite. Q est le nombre de pixels limites pour la région correspondante.

B = bwboundaries (BW, conn) spécifie la connectivité à utiliser lors du traçage des frontières parent et enfant. Conn peut avoir l'une des valeurs scalaires suivantes.

**label2rgb**

RGB = label2rgb (L) convertit une matrice d'étiquettes, L, comme celles renvoyées par labelmatrix, bwlabel, bwlabeln ou bassin versant, en une image couleur RVB pour visualiser les régions marquées. La fonction label2rgb détermine la couleur à attribuer à chaque objet en fonction du nombre d'objets dans la matrice d'étiquettes et de la plage de couleurs dans la carte de couleurs. La fonction label2rgb sélectionne les couleurs de toute la plage.

RGB = label2rgb (L, map) définit la carte de la carte de couleurs à utiliser dans l'image RVB. Map peut avoir l'une des valeurs suivantes:

* Matrice de carte de couleur n-par-3
* Chaîne contenant le nom d'une fonction de colorimétrie MATLAB, telle que 'jet' ou 'gray' (Voir la carte des couleurs pour une liste des cartes de couleurs prises en charge.)
* Poignée de fonction d'une fonction de colormap, telle que @jet ou @gray

Si vous ne spécifiez pas de carte, la valeur par défaut est 'jet'.

RGB = label2rgb (L, map, zerocolor) définit la couleur RVB des éléments étiquetés 0 (zéro) dans la matrice d'étiquette d'entrée L. En tant que valeur de zerocolor, spécifiez un triple RVB ou l'une des chaînes répertoriées dans ce tableau.

**boundary**

[P, q] = boundary (obj) renvoie les points limites entre les segments de l'objet de distribution par morceaux, obj. P est un vecteur de probabilités cumulatives à chaque frontière. Q est un vecteur de quantiles à chaque frontière.

[P, q] = boundary (obj, i) renvoie p et q pour la ième frontière.

**edgetaper**

J = edgetaper (I, PSF) floue les bords de l'image d'entrée I en utilisant la fonction d'étalement de points PSF. La taille du PSF ne peut pas dépasser la moitié de la taille de l'image dans n'importe quelle dimension.

L'image de sortie J est la somme pondérée de l'image originale I et de sa version floue. La matrice de pondération, déterminée par la fonction d'autocorrélation de PSF, fait J égal à I dans sa région centrale et égale à la version floue de I près des bords.

La fonction edgetaper réduit l'effet de sonnerie dans les méthodes de décoloration d'image qui utilisent la transformée de Fourier discrète, comme deconvwnr, deconvreg et deconvlucy.

**Imageinfo**

Imageinfo crée un outil d'information d'image associé à l'image dans la figure actuelle. L'outil affiche des informations sur les attributs de base de l'image cible dans une figure distincte. Imageinfo obtient des informations sur les attributs d'image en interrogeant le CData de l'objet image.

Wiener2 passe-bas filtre une image en niveaux de gris qui a été dégradée par un bruit additif de puissance constant. Wiener2 utilise une méthode Wiener adaptative pixelwise basée sur des statistiques estimées à partir d'un voisinage local de chaque pixel.J = wiener2 (I, [m n], noise) filtre l'image I en utilisant le filtrage Wiener adaptatif pixelwise, en utilisant des quartiers de taille m-by-n pour estimer la moyenne d'image locale et l'écart-type. Si vous omettez l'argument [m n], m et n par défaut sont 3. La puissance du bruit additif (bruit blanc gaussien) est supposée être du bruit.J, noise] = wiener2 (I, [m n]) estime également la puissance du bruit additif avant de faire le filtrage. Wiener2 renvoie cette estimation en bruit

La fonction wiener2 applique un filtre Wiener (un type de filtre linéaire) à une image de manière adaptative, en s'adaptant à la variance d'image locale. Lorsque la variance est importante, wiener2 réalise peu de lissage. Lorsque la variance est faible, wiener2 effectue plus de lissage.

Cette approche produit souvent de meilleurs résultats que le filtrage linéaire. Le filtre adaptatif est plus sélectif qu'un filtre linéaire comparable, préservant les bords et d'autres parties haute fréquence d'une image. En outre, il n'ya pas de tâches de conception; La fonction wiener2 gère tous les calculs préliminaires et met en œuvre le filtre pour une image d'entrée. Wiener2, cependant, nécessite plus de temps de calcul que le filtrage linéaire.

Wiener2 fonctionne mieux lorsque le bruit est constant («blanc») bruit additif, comme le bruit gaussien. L'exemple ci-dessous s'applique wiener2 à une image de Saturne qui a eu un bruit gaussien ajouté.

1-Ead dans une image. Parce que l'image est une image truecolor, l'exemple le convertit en niveaux de gris

2-L'exemple ajoute alors du bruit gaussien à l'image, puis affiche l'image. Parce que l'image est assez grande, la figure ne montre qu'une partie de l'image

3-Supprimez le bruit en utilisant la fonction wiener2. Encore une fois, la figure ne montre qu'une partie de l'image

H = fspecial (type) crée un filtre bidimensionnel h du type spécifié. Fspecial renvoie h comme noyau de corrélation, qui est la forme appropriée à utiliser avec imfilter. Type est une chaîne ayant une de ces valeurs

H = fspecial ('disque', rayon) renvoie un filtre de moyenne circulaire (pillbox) dans la matrice carrée du côté 2 \* rayon + 1. Le rayon par défaut est

gaussian:: H = fspecial ('gaussien', hsize, sigma) renvoie un filtre passe-bas gaussien symétrique en rotation de taille hsize avec un écart type sigma (positif). Hsize peut être un vecteur spécifiant le nombre de lignes et de colonnes dans h, ou il peut être un scalaire, auquel cas h est une matrice carrée. La valeur par défaut pour hsize est [3 3]; La valeur par défaut de sigma est 0,5

H = fspecial ('laplacien', alpha) renvoie un filtre 3-par-3 approximant la forme de l'opérateur laplacien bidimensionnel. Le paramètre alpha contrôle la forme du Laplacien et doit être compris entre 0,0 et 1,0. La valeur par défaut pour alpha est 0,2.

H = fspecial ('log', hsize, sigma) renvoie un laplacien symétrique de rotation du filtre gaussien de taille hsize avec un écart type sigma (positif). Hsize peut être un vecteur spécifiant le nombre de lignes et de colonnes dans h, ou il peut être un scalaire, auquel cas h est une matrice carrée. La valeur par défaut pour hsize est [5 5] et 0,5 pour sigma

H = fspecial ('motion', len, theta) renvoie un filtre pour rapprocher, une fois convolué avec une image, le mouvement linéaire d'une caméra par len pixels, avec un angle de theta degrés dans le sens inverse des aiguilles d'une montre. Le filtre devient un vecteur pour les mouvements horizontaux et verticaux. Le len par défaut est 9 et la valeur par défaut theta est 0, ce qui correspond à un mouvement horizontal de neuf pixels.

Pour calculer les coefficients du filtre, h, pour le «mouvement»:

1. Construire un segment de ligne idéal avec la longueur et l'angle désirés, centré au coefficient de centre de h.

2. Pour chaque emplacement de coefficient (i, j), calculez la distance la plus proche entre cet emplacement et le segment de droite idéal.

3.h = max (1 - la plus proche\_distance, 0);

4. Normalize h: h = h / (somme (h (:))

H = fspecial ('prewitt') renvoie le filtre 3-by-3 h (illustré ci-dessous) qui met l'accent sur les bords horizontaux en rapprochant un gradient vertical. Si vous devez accentuer les bords verticaux, transposez le filtre h '.

[1 1 1

0 0 0

-1 -1 -1]

H = fspecial ('sobel') renvoie un filtre de 3 par 3 h (illustré ci-dessous) qui met l'accent sur les bords horizontaux en utilisant l'effet de lissage en approchant un gradient vertical. Si vous devez accentuer les bords verticaux, transposez le filtre h '.

[1 2 1

0 0 0

-1 -2 -1]

H = fspecial ('unsharp', alpha) renvoie un filtre d'amélioration du contraste de 3 par 3. Fspecial crée le filtre unsharp à partir du négatif du filtre laplacien avec le paramètre alpha. Alpha contrôle la forme du Laplacien et doit être dans la plage de 0,0 à 1,0. La valeur par défaut pour alpha est 0,2

IDX = kmeans (X, k) partitionne les points dans la matrice de données n-by-p X en k clusters. Ce partitionnement itératif minimise la somme, sur tous les clusters, des sommes intra-cluster des distances centroïdes point à cluster. Les rangées de X correspondent aux points, les colonnes correspondent aux variables. Kmeans renvoie un vecteur n-par-1 IDX contenant les indices de cluster de chaque point. Par défaut, kmeans utilise des distances euclidiennes carrées. Lorsque X est un vecteur, kmeans le considère comme une matrice de données n-par-1, quelle que soit son orientation. [IDX, C] = kmeans (X, k) renvoie les k localisations du centroïde du cluster dans la matrice k-by-p C. [IDX, C, sumd] = kmeans (X, k) renvoie les sommes intra-cluster des distances point à centre dans le vecteur 1-by-k sumd. [IDX, C, sumd, D] = kmeans (X, k) renvoie des distances de chaque point à chaque centroïde dans la matrice D. [...] = kmeans (..., param1, val1, param2, val2, ...) vous permet de spécifier des paires de paramètres / valeurs optionnelles pour contrôler l'algorithme itératif utilisé par kmeans

I = mat2gray (A, [amin amax]) convertit la matrice A en l'image d'intensité I. La matrice renvoyée I contient des valeurs comprises entre 0,0 (noir) et 1,0 (pleine intensité ou blanc). Amin et amax sont les valeurs de A qui correspondent à 0,0 et 1,0 dans I.

I = mat2gray (A) fixe les valeurs de amin et amax aux valeurs minimale et maximale dans A

Level = graythresh (I) calcule un seuil global (niveau) qui peut être utilisé pour convertir une image d'intensité en une image binaire avec im2bw. Niveau est une valeur d'intensité normalisée qui se trouve dans la plage [0, 1].

La fonction graythresh utilise la méthode d'Otsu, qui choisit le seuil pour minimiser la variance intraclasse des pixels noir et blanc.

Les matrices multidimensionnelles sont converties automatiquement en matrices 2-D à l'aide de remodeler. La fonction graythresh ignore toute partie imaginaire non nulle de I.

[Niveau EM] = graythresh (I) renvoie la métrique d'efficacité, EM, comme second argument de sortie. La métrique d'efficacité est une valeur dans la plage [0 1] qui indique l'efficacité du seuillage de l'image d'entrée. La limite inférieure n'est atteignable que par des images ayant un seul niveau de gris, et la limite supérieure n'est atteignable que par des images à deux valeurs

B = remodelage (A, m, n) (reshape) renvoie la matrice m-by-n B dont les éléments sont pris par colonne de A. Une erreur se produit si A n'a pas m \* n éléments

[Center, U, obj\_fcn] = fcm (données, cluster\_n) applique la méthode fuzzy c-means à un ensemble de données donné.

Les arguments d'entrée de cette fonction sont

Données: ensemble de données à regrouper; Chaque ligne est un point de données échantillon

Cluster\_n: nombre de clusters (plus d'un)

Les arguments de sortie de cette fonction sont

Centre: matrice des centres de clusters finaux où chaque ligne fournit les coordonnées du centre

U: matrice de partition floue finale (ou matrice de fonction d'appartenance)

Obj\_fcn: valeurs de la fonction objective pendant les itérations

Fcm (data, cluster\_n, options) utilise une variable d'argument supplémentaire, options, pour contrôler les paramètres de clustering, introduire un critère d'arrêt, définir l'affichage des informations d'itération ou les deux.

Options (1): exposant pour la matrice de partition U (par défaut: 2.0)

Options (2): nombre maximum d'itérations (par défaut: 100)

Options (3): montant minimum d'amélioration (par défaut: 1e-5)

Options (4): affichage des informations pendant l'itération (par défaut: 1)

B = sort (A) trie les éléments le long des différentes dimensions d'un tableau et arrange ces éléments par ordre croissant.

Erreur ('msgIdent', 'msgString', v1, v2, ..., vN) génère une exception si la fonction en cours d'exécution teste et confirme une condition défectueuse ou inattendue. En fonction de la façon dont le programme a été conçu pour répondre à l'erreur, MATLAB soit entre un bloc catch pour gérer la condition d'erreur ou quitte le programme.

L'argument msgIdent est une chaîne d'identificateur de message unique que MATLAB attache au message d'erreur lorsqu'il lance l'erreur. Un identificateur de message a le composant de format: mnémonique. Son but est de mieux identifier la source de l'erreur (voir Message Identifiers dans la documentation MATLAB Programming Fundamentals pour plus d'informations).

J = imadjust(I) maps the intensity values in grayscale image I to new values in J such that 1% of data is saturated at low and high intensities of I. This increases the contrast of the output image J. This syntax is equivalent to imadjust(I,stretchlim(I)).

D = bwdist (BW) calcule la transformée de distance euclidienne de l'image binaire BW. Pour chaque pixel en BW, la transformée de distance affecte un nombre qui est la distance entre ce pixel et le pixel non nul le plus proche de BW. Bwdist utilise la métrique distance euclidienne par défaut. BW peut avoir n'importe quelle dimension. D est de la même taille que BW.

L = watershed(A) computes a label matrix identifying the watershed regions of the input matrix A, which can have any dimension. The elements of L are integer values greater than or equal to 0. The elements labeled 0 do not belong to a unique watershed region. These are called watershed pixels. The elements labeled 1 belong to the first watershed region, the elements labeled 2 belong to the second watershed region, and so on.

By default, watershed uses 8-connected neighborhoods for 2-D inputs and 26-connected neighborhoods for 3-D inputs. For higher dimensions, watershed uses the connectivity given by conndef(ndims(A),'maximal').

IM2 = imtophat (IM, SE) effectue un filtrage morphologique du chapeau supérieur sur l'image d'entrée binaire IM ou en niveaux de gris. Le filtrage hiérarchique calcule l'ouverture morphologique de l'image (en utilisant imopen) puis soustrait le résultat de l'image originale. Imtophat utilise l'élément structurant SE, où SE est renvoyé par strel. SE doit être un objet d'élément structurant unique, et non pas un tableau contenant plusieurs objets d'éléments structurants.

IM2 = imtophat (IM, NHOOD) où NHOOD est un tableau de 0s et 1s qui spécifie la taille et la forme de l'élément structurant, est le même que imptophat (IM, strel (NHOOD)).

%Un très utile morphologique trop soustraire l'arrière-plan

View La position du spectateur (le point de vue) détermine l'orientation des axes. Vous spécifiez le point de vue en termes d'azimut et d'élévation, ou par un point dans l'espace tridimensionnel.

View (az, el) et view ([az, el]) définissent l'angle de vue d'un tracé tridimensionnel. L'azimut, az, est la rotation horizontale autour de l'axe z mesurée en degrés à partir de l'axe des y négatif. Les valeurs positives indiquent la rotation dans le sens contraire des aiguilles d'une montre du point de vue. El est l'élévation verticale du point de vue en degrés. Les valeurs positives d'élévation correspondent au déplacement au-dessus de l'objet; Les valeurs négatives correspondent au déplacement sous l'objet.

La fonction imagesc évalue les données d'image dans la gamme complète de la carte de couleurs courante et affiche l'image. (Voir Exemples pour une illustration.)

Imagesc (C) affiche C comme image. Chaque élément de C correspond à une zone rectangulaire dans l'image. Les valeurs des éléments de C sont des indices dans la carte de couleur courante qui déterminent la couleur de chaque patch

Une courbe de contour affiche les isolignes de la matrice Z. Étiquettez les courbes de niveau à l'aide de clabel.

Le contour (Z) trace un tracé de contour de la matrice Z, où Z est interprété comme des hauteurs par rapport au plan x-y. Z doit être au moins une matrice 2-par-2 qui contient au moins deux valeurs différentes. Le nombre de lignes de contour et les valeurs des courbes de niveau sont choisis automatiquement en fonction des valeurs minimale et maximale de Z. Les plages des axes x et y sont [1: n] et [1: m], où [ M, n] = taille (Z).

Le contour (Z, n) trace un tracé de contour de la matrice Z avec n niveaux de contour

R = randn (n) renvoie une matrice n-par-n contenant des valeurs pseudo-aléatoires tirées de la distribution normale standard. Randn (m, n) ou randn ([m, n]) renvoie une matrice m-by-n. Randn (m, n, p, ...) ou randn ([m, n, p, ...]) renvoie un tableau m-by-n-by-p-by -... Randn retourne un scalaire. Randn (taille (A)) renvoie un tableau de la même taille que A

BW2 = imfill (BW, 'trous') remplit les trous dans l'image binaire BW. Un trou est un ensemble de pixels d'arrière-plan qui ne peuvent être atteints en remplissant le fond du bord de l'image.

I=im2bw(Img);

figure,imshow(I), titlle(‘image originale’)

Il=u(:,:,1);

figure,imshow(Il) , titlle(‘image selon x’)

Il1=u(:,:,2);

figure,imshow(Il1) , titlle(‘image selon y’)

Il=im2bw(Il);

IL1=~Il;

figure,imshow(IL1) , titlle(‘image selon x inversé’)

Il1=im2bw(Il1);

IL2=~Il1;

figure,imshow(IL2) , titlle(‘image selon y inversé’)

ll=I-IL1;

figure,imshow(ll) , titlle(‘image moins x’)

ll1=I-IL2;

figure,imshow(ll) , titlle(‘image moins y’)

label= bwlabel(j, 8); % etiquetation des regions avec bwlabel

couleurs\_labels= label2rgb (label, 'hsv', 'k', 'shuffle');% couleurs aleatoires pour les labels

IM2 = imerode (IM, SE) érode l'image binaire, binaire, ou binaire IM, renvoyant l'image érodée IM2. L'argument SE est un objet élément structurant ou un tableau d'objets élément structurant renvoyés par la fonction strel

Glcm = graycomatrix (I) crée une matrice de cooccurrence de niveau de gris (GLCM) à partir de l'image I. graycomatrix crée le GLCM en calculant la fréquence avec laquelle un pixel à intensité de gris (intensité de gris) i est horizontalement adjacent à un pixel Valeur j. Chaque élément (i, j) dans glcm spécifie le nombre de fois où le pixel avec la valeur i s'est produit horizontalement à côté d'un pixel de valeur j.

Graycomatrix calcule le GLCM à partir d'une version mise à l'échelle de l'image. Par défaut, si I est une image binaire, graycomatrix réduit l'image à deux niveaux de gris. Si I est une image d'intensité, graycomatrix calcule l'image à huit niveaux de gris. Vous pouvez spécifier le nombre de niveaux de gris graycomatrix utilise pour l'échelle de l'image en utilisant le paramètre 'NumLevels', et la façon dont graycomatrix échelles les valeurs en utilisant le paramètre 'GrayLimits' - voir Paramètres.

La figure suivante montre comment la méthode graycomatrix calcule plusieurs valeurs dans le GLCM de l'image 4 par 5 I. L'élément (1,1) du GLCM contient la valeur 1 car il n'y a qu'une seule occurrence dans l'image où deux pixels adjacents horizontalement Ont les valeurs 1 et 1. L'élément (1,2) dans le GLCM contient la valeur 2 car il ya deux instances dans l'image où deux pixels adjacents horizontalement ont les valeurs 1 et 2. graycomatrix continue ce traitement pour remplir toutes les valeurs Dans le GLCM