Infographie

A. Sere
University Nazi BONI

Université Virtuelle Africaine

INFORMATIQUE APPLIQUÉE: CSI 4203

L'INFOGRAPHIE

Dr. Abdoulaye Sere

Avant-propos

L'Université Virtuelle Africaine (UVA) est fière de participer à accès à l'éducation dans les pays africains en produisant du matériel d'apprentissage de qualité. Nous sommes également fiers de contribuer à la connaissance globale, pour nos ressources éducatives sont principalement accessibles de l'extérieur du continent africain.

Ce module a été développé dans le cadre d'un programme de diplôme et diplôme en informatique appliquée, en collaboration avec 18 institutions partenaires dans 16 pays africains. Un total de 156 modules ont été développés ou traduits pour assurer la disponibilité en anglais, français et portugais. Ces modules sont également disponibles en tant que ressources éducatives ouvertes (OER) à oer.avu.org.

Au nom de l'Université Virtuelle Africaine et notre patron, nos institutions partenaires, la Banque africaine de développement, je vous invite à utiliser ce module dans votre établissement, pour leur propre éducation, partager aussi largement que possible et participer activement aux communautés AVU de pratique d'intérêt. Nous nous engageons à être à l'avant-garde du développement et de partage ouvert de ressources pédagogiques.

L'Université Virtuelle Africaine (UVA) est une organisation intergouvernementale panafricaine mis en place par lettre recommandée avec un mandat d'augmenter l'accès à l'enseignement supérieur et de formation de qualité grâce à l'utilisation novatrice des technologies de communication de l'information. Une charte instituant la UVA Organisation intergouvernementale, signée à ce jour par dix-neuf (19) Les gouvernements africains - Kenya, Sénégal, Mauritanie, Mali, Côte d'Ivoire, Tanzanie, Mozambique, République démocratique du Congo, Bénin, Ghana, République de Guinée, le Burkina Faso, le Niger, le Soudan du Sud, le Soudan, la Gambie, la Guinée-Bissau, l'Ethiopie et le Cap-Vert.

Les institutions suivantes ont participé au programme informatique appliquée: (1) Université d'Abomey Calavi au Bénin; (2) University of Ougagadougou au Burkina Faso; (3) Université Lumière Bujumbura Burundi; (4) Université de Douala au Cameroun; (5) Université de Nouakchott en Mauritanie; (6) Université Gaston Berger Sénégal; (7) Université des Sciences, Techniques et Technologies de Bamako au Mali (8) Institut de la gestion et de l'administration publique du Ghana; (9) Université des sciences et de la technologie Kwame Nkrumah au Ghana; (10) Université Kenyatta au Kenya; (11) Université Egerton au Kenya; (12) Université d'Addis-Abeba en Ethiopie (13) Université du Rwanda; (14) University of Salaam en Tanzanie Dar; (15) Université Abdou Moumouni Niamey Niger; (16) Université Cheikh Anta Diop au Sénégal; (17) Université pédagogique au Mozambique; E (18) L'Université de la Gambie en Gambie.

Bakary Diallo

le Recteur

Université Virtuelle Africaine

Crédits de Production

Auteur

Abdoulaye Sere

Pair Réviseur

Luc Ngend

UVA – Coordination Académique

Dr. Marilena Cabral

Coordinateur global Sciences Informatiques Apliquées

Prof Tim Mwololo Waema

Coordinateur du module

Florence Tushabe

Concepteurs pédagogiques

Elizabeth Mbasu

Benta Ochola

Diana Tuel

Equipe Média

Sidney McGregor

Barry Savala

Edwin Kiprono

Kelvin Muriithi

Victor Oluoch Otieno

Michal Abigael Koyier

Mercy Tabi Ojwang

Josiah Mutsogu

Kefa Murimi

Gerisson Mulongo

Droits d'auteur

Supporté par

Projet Multinational II de l'UVA financé par la Banque africaine de développement.

Table des matières

Aperçu du cours

Bienvenue au cours d'Infographie

L'infographie est l'étude de la création, de la manipulation et de l'utilisation d'images dans l'ordinateur. Les applications de l'Infographie sont nombreuses telles que les divertissements (Films, effets spéciaux, Films d'animation, Jeux vidéos), la science et la technologie (conception assistée par ordinateur, visualisation scientifique), les simulateurs (de vol, etc...), les arts graphiques (photoshop, illustrator, GIMP).

Ce cours permettra d'avoir des connaissances en traitement d'images comment synthétiser ou créer une image et comment manipuler ou rétoucher des images en dimension 2 et 3 en utilisant des opérateurs disponibles dans un logiciel de traitement d'images. Ainsi une base théorique avec les concepts de base décrivant les élements structurants une image et la définition théorique des opérateurs et des cas pratiques sont proposées dans ce cours.

Prérequis

1. Avoir des notions de géométrie analytique et de Mathématiques
2. Maîtrise des concepts orientés objet et de la programmation orientée objet
3. Bureautique (traitement de texte)

Matériaux

Les matériaux nécessaires pour compléter ce cours comprennent:

1. Une librairie permettant de définir des images en dimension 2 et 3 et d'appeler les fonctions disponibles ou de construire de nouvelles fonctions telles que OPENCV, CIMG. H ;
2. Un IDE prenant en compte un langage de programmation orientée objet comme C++. Les IDE comme CODEBLOCK, DEVCPP, libre d'accès peuvent être utilisées ; D'autre langage de programmation comme MathLab peut être étudié également en perspective
3. Un logiciel de traitement d'images doit être utilisé pour montrer une application des concepts présentés théoriquement comme GIMP, POVRAY.
4. Une machine qui possède une carte graphique ou un processeur graphique intégré dans le CPU permettant la création, la visualisation et le traitement des images.

Objectifs du cours

À la fin de ce cours, l›étudiant devrait :

1. Acquérir des connaissances sur les élements de la géométrie discrète ;
2. Comprendre comment segmenter ou filtrer une image ;
3. Connaître les transformations fournit par une bibliothèque (OPENGL, OPENCV, CIMG.H…) de la programmation s'appliquant aux images en dimension 2 ou 3
4. Être en mesure d'utiliser un logiciel de traitement d'images en mettant en œuvre des transformations.

Unités

Unité 0: Rappel de géométrie analytique et de programmation orientée objet

Cette unité permet de faire un rappel des notions de géométrie analytique permettant de mieux comprendre les autres unités.

De plus, cette unité rappelle les concepts orientés objet (application en C++) qui sont utiles dans l'apprentissage d'une bibliothèque (OPENGL, OPENCV ou CIMG.h…)

Unité 1: Notion d'une image et élements de géométrie discrète

Cette unité est un cours théorique sur les primitives discrètes (droite discrète comme la droite discrète de Brésenham, le cercle discret ou plus généralement les hyperplans discrets et les hypersphères discrètes) .

Unité 2: Segmentation, filtrage et morphologie (Erosion, Dilatation…)

Cette unité est un cours théorique qui présente les techniques de segmentation, de filtrage et de morphologie

Unité 3: Transformations des images

Cette unité est un cours théorique consacrée à l'étude des transformations comme les rotations, les translations, les hométhéties. La transformée de Hough sera également présentée.

Unité 4: Etude de cas sur des images 2D et 3D

Cette unité permet à l'étudiant de créér et de manipuler des images 2D et 3D en utilisant une bibliothèque (OPENCV ou CIMG.h…). Un logiciel de traitement (POVRAY, GIMP...) est également utilisé pour permettre à l'étudiant de travailler sur des aspects pratiques.

Évaluation

Les évaluations formatives (vérification de progrès) sont inclus dans chaque unité.

Les évaluations sommatives (tests et travaux finaux) sont fournies à la fin de chaque module et traitent des connaissances et compétences du module.

Les évaluations sommatives sont gérés à la discrétion de l'établissement qui offre le cours. Le plan d'évaluation proposé est le suivant:

1	Devoir individuel sur table	20%
2	Projet par équipe	20%
3	Examen final	60%

Plan

Unité	Sujets et Activités	Durée estimée
Notion de géométrie analytique et de programmation orientée objet	Notion de géométrie analytique Trigonométrie ; Coordonnées polaires en dimension 2; Coordonnées en dimension 3 ; Points, vecteurs, Matrices Transformations : symétrie, homothétie, rotation, similitudes. Hyperplan, hypersphère, parabole ; Distance d'un point à un hyperplan. Programmation orientée objet Concept objet Encapsulation : classe, propriétés, méthode, objet, règle d'accès private, public, protected Abstraction : classe abstraite Héritage : héritage simple, héritage multiple; Polymorphisme	10 h
Notion d'une image et élements de géométrie discrète	Notions d'images numérique Notion d'élements structurants Hyperplan discret : droite discrète de réveillès, droite discrète de Bresenham. Hypersphère discrète : cercle discret Courbes discrètes quelconques	10 h

Segmentation, filtrage et morphologie des images	Segmentation Filtres Morphologie	8 h
Transformations des images	Translation Rotation Homothétie Projection Transformées de Hough	8 h
Etudes de cas sur des images 2D et 3D	Introduction à OpenCV (ou OPENGL ou CIMG.h) Etude des types de données Etude des Fonctions 2D et 3D Etude d'un de logiciel de traitement d'images (GIMP ou autre logiciel, …)	30 h

Unité 0: Évaluation Diagnostique

Introduction à l'unité

Cette unité vous permettra de vérifier les connaissances que vous devez avoir avant de commencer le cours. Vous pouvez faire l'évaluation de l'unité avant de faire des activités d'apprentissage pour aider à rafraîchir vos connaissances.

Ainsi, cette unité permettra de faire un rappel des notions de géométrie analytique.

Objectifs de l'unité

À la fin de cette unité, vous devriez être capable de:

Maîtriser des notions sur les différents techniques de representation d'un point dans l'espace en dimension 2, en dimension 3 et en dimension n.

Comprendre la définition des transformations comme la rotation, la translation, l'homothétie, la symétrie centrale ou la symétrie orthogonale et plus généralement la notion de similitude.

Comprendre la definition des courbes de l'espace et des surfaces (hyperplan, hypersphère, parabole, Conique).

Comprendre la notion de distance d'un point à un hyperplan.

Termes clés

Coordonnées d'un point: Une représentation d'un point dans un repère

Évaluation de l'unité

Vérifiez votre compréhension!

Exercice 1 (Géométrie analytique)

Définir transformations avec un exemple : rotation, translation, homothétie;

Donner la définition d'un hyperplan avec un exemple;

Donner la définition d'une hypersphère avec un exemple ;

Donner un exemple d'héritage multiple en C++

Exercice 2 (Géométrie analytique)

On considère les transformations géométriques suivantes en 2D:

T1 : symétrie par rapport à l'axe des abscisses

T2 : rotation de 30° autour du point (2,2)

T3 : translation de vecteur directeur (-4,2)

T4 : dilatation $x' = 2x$, $y' = 3y$

1. Donner, en coordonnées homogènes, les matrices correspondantes.

2. On effectue à la suite les transformations T1, T2, T3, T4. Quelle est la matrice M de la transformation globale ?

Exercice 3 (classes, héritage)

Définir les termes suivants : classe, objet, polymorphisme, encapsulation, héritages;

Proposer une classe personne en C++ comprenant les propriétés comme nom, prenom de type string ;

Proposer une classe Etudiant qui hérite de cette classe Personne. La classe Etudiant comporte une propriété matricule de type int;

Donner un exemple d'héritage multiple en C++

Lectures et autres ressources

Les lectures et ressources de cette unité sont se trouvent au niveau des lectures et autres ressources du cours.

Unité 1: Notion D'une Image et Éléments de Géométrie Discrète

Introduction à l'unité

Dans cette unité, nous définissons les types d'images avec leurs avantages et leurs inconvénients, les élements structurants d'une image, les primitives discrètes telles que la droite discrète de Bresenham, le cercle discret ou généralement les hyperplans discrets et les hypersphères discrètes. Nous présentons également des polygones, des courbes qui sont souvent utilisées dans l'élaboration d'un dessin.

Objectifs de l'unité

À la fin de cette unité, vous devriez être capable de:

Savoir choisir un format d'images ;

1. Maîtriser la définition mathématique des primitives discrètes comme les droites discrètes, les cercles discrets et plus générallement les hyperplans discrets et les hypersphères discrètes;
2. Pouvoir écrire un algorithme de tracé d'une primitive discrète à partir de sa définition mathématique ;

Termes clés

Pixel : un pixel est l'élement de base d'une image

Résolution : la résolution indique le nombre de pixel dans un champ

Droite discrète : une droite discrète est un ensemble de points discrets alignés sur une droite continue

Cercle discret : un cercle discret est un ensemble de points discrets alignés sur un cercle continu

Image matricielle: une image matricielle est un tableau de points

Image vectorielle: une image vectorielle est une image définie sur la base d'une équation mathématique

Activités d'apprentissage

Activité 1 Types d'images et élements structurants d'une image

Introduction

Dans cette activité, nous allons présenter les différents types d'images avec leurs avantages et leurs inconvénients. Les élements structurants et des techniques de pavage d'une image seront évoqués.

Détails de l'activité

Types d'images

Les images numériques sont acquises par des capteurs comme les appareils photonumériques, les camescopes ou par numérisation par un scanner.

Il existe principalement deux types d'images : les images matricielles et les images vectorielles.

Les images matricielles

Une image matricielle est une image dont les données peuvent être representées par une matrice. En dimension deux, un point lumineux est appéllé un pixel. Le pixel est dans ce cas réprésenté dans une repère par ses coordonnées (x, y). On peut de cette façon dérouler un algorithme pour traiter chaque pixel. Ll'opération de digitalisation ou d'acquisition ou encore de numérisation, consiste à transformer en un nombre fini de points élémentaires les points lumineux d'une image.

Ces points codés sont rangés en lignes et en colonnes avec la correspondance un élément de la matrice correspond à un point de l'écran de l'ordinateur.

La figure 1 montre un exemple de matrice 5 lignes x 6 colonnes. Chaque cellule correspond à un point lumineux, appéllé pixel.

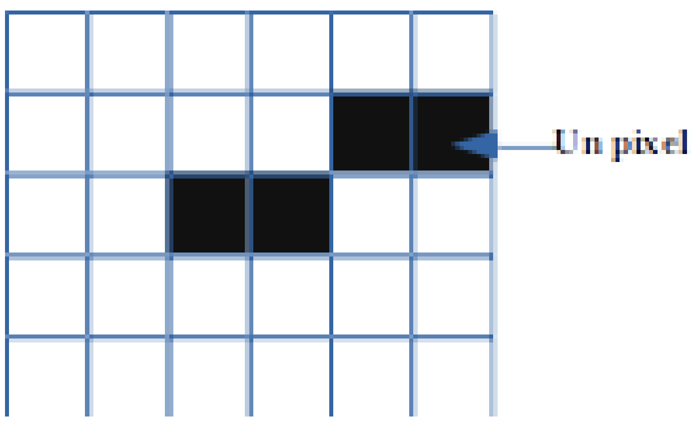


Figure 1 : Une image matricielle 5 lignes x 6 colonnes, montrant un ensemble de 4 pixels allumés formant un segment de droite discrète.

Dans une image en couleur en dimension 2, comme dans la figure 2 portant sur une image classique de lena, un pixel est associé à une combinaison de couleur.

Figure 2 : Une image classique de lena avec des pixels en couleur.

Dans une image, les pixels n'ont pas toujours la même intensité.

Les avantages principaux de l'image matricielle

Les mécanismes d'affichage sont simples du fait du codage des points. Les dispositifs principaux de visualisation sont d'ailleurs des dispositifs qui affichent des suites de points (balayage). L'application des algorithmes de traitement d'image est facilitée par la structure matricielle de l'image La conversion de format est également facilitée puisqu'elle se résume à une transformation matricielle.

Chaque partie d'une image matricielle est, en principe, indépendante d'une autre partie. Par la suite une dégradation d'une partie de l'image n'entraîne pas la dégradation totale de l'image.

Les inconvénients principaux des images matricielles

L'image matricielle contient un nombre fini de points appelé résolution (nombre de points par ligne x nombre de points par colonne). Un dispositif de restitution contient, lui-aussi un nombre fixé de points qui constitue une résolution généralement différente de la première.

Les modifications spatiales d'une image matricielle, simples dans leur principe, impliquent toutefois un temps de calcul important et présentent certains problèmes : perte d'information par réduction de taille, effets d' escalier.

Dans une image matricielle, le nombre de points à traiter est très grand, ce qui rend nécessaire l'emploi de processeurs adaptés.

Codage d'images matricielles

Les images binaires (pixels en noir ou en blanc) sont des images ou un pixel est codé sur un bit : 0 pour représenter la couleur noire et 1 pour la couleur blanche. Une image peut être représente en couleur de niveaux de gris comme dans la figure ci-dessous :

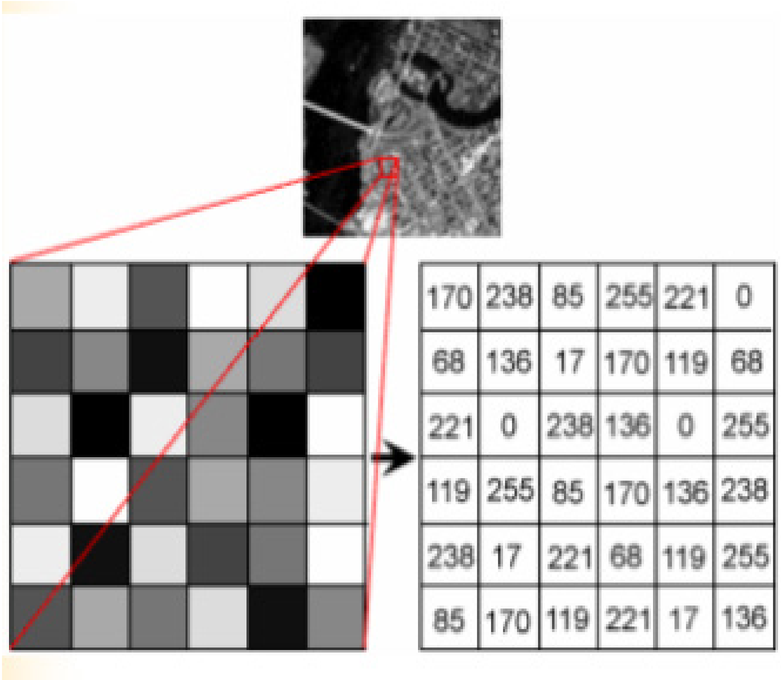


Figure 3 : un exemple de codage d'une image à niveau de gris (0 à 255)

Les images en couleur : la différence entre une image en niveaux de gris et une image en couleur se trouve notamment du nombre réservé pour coder un point lumineux ou un pixel. Par exemple un triplet (a, b,c) correspondant respectivement aux couleurs (R,G, B) (Le modèle RVB ou RGB (Rouge, Vert, Bleu)) peut être utilisé pour repésenté la valeur d'un pixel : chaque couleur est codé sur 8 bits soit une valeur allant de 0 à 255. Suivant la variante de couleur, un pixel peut être représenté sur 8 bits, 16 bits, 32 bits. D'autres couleurs sont obtenus par la combinaison de ces trois couleurs comme le montre la figure ci-dessous. En outre, ils existent d'autres modèles de couleurs tels que : le modèle CMJ ou CMY (Cyan, Magenta, Jaune), le modèle TSL (Teinte, Saturation, Luminance) ou HSV (Hue, Saturation, Value)

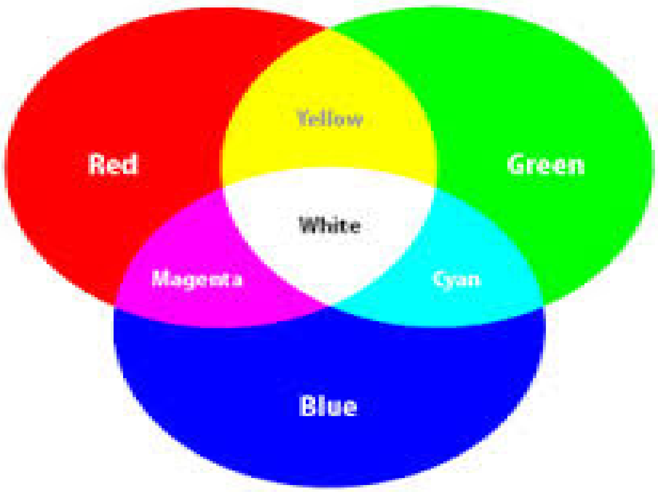


Figure 4: l'espace de couleurs

[Source: http://07268grum.wordpress.com/2013/11/08/colour-space-greyscale-rgb-yuv-lumiance-chrominance-hsv-hue-saturation-value/]

En infographie, il est possible d'appliquer des opérateurs noir et blanc pour convertir une image en couleur en une image noir et blanc comme le montre la figure suivante :

Figure 5: une image noir blanc et une image en couleur

[Source: http://www.learnquebec.ca/en/content/professional_development/media_literacy/literacy-today-classroom/photo-language-photograph.html]

Format d'images matricielles

Nous présentons ici quelques formats d'images matricielles dans le tableau ci-dessous:

Format	Description
BMP	origine Microsoft pour Windows 3.x; environnement PC; de plus en plus répandu
PCX	origine Paintbrush de Z-soft; utilisé en environnement PC; permet de traiter des images 8 bits; non adapté aux images 16, 24, 32 bit; algorithme de compression : RLC
GIF	assez répandu; d'origine Compuserve; utilisé dans un environnement PC; codage effectué sur 8 bits (256 couleurs); algorithme de compression : LZW

TIFF	Origine Aldus et Microsoft, pour les images scannées; utilisé en environnement PC et Mac; plusieurs algorithmes de compression : RLC, LZW
PICT	Format de base de QuickDraw de Mac; traite aussi le vectoriel; environnement Mac
TGA	origine Truevision; algorithme de compression RLC; environnement PC
FAX	utilisé pour la transmission de documents (télécopie); codage binaire; application d'un codage RLC puis d'un codage de Huffman

Exemple d'image matricielle

La figure suivante donne un exemple d'images bitmap.

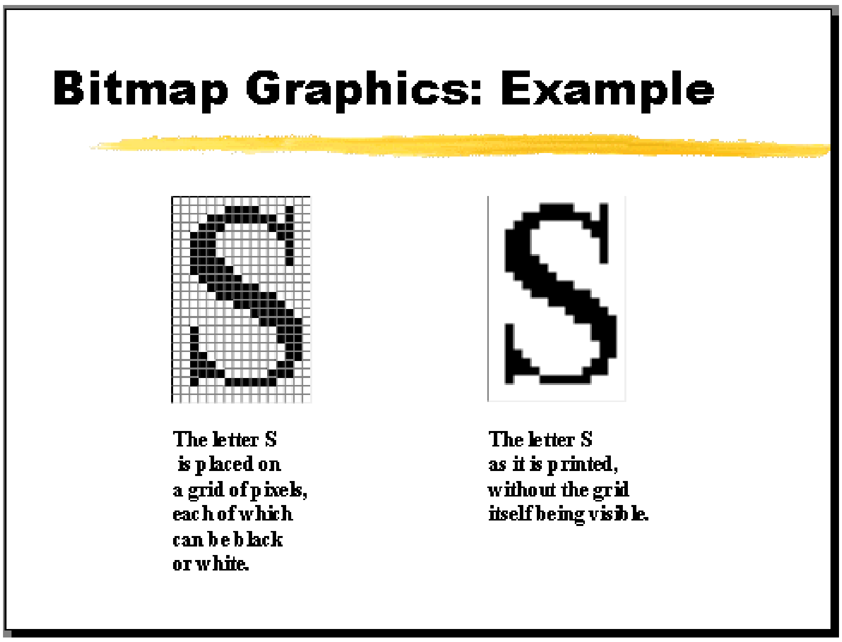


Figure 6: Bitmap Image

[Source: http://ecee.colorado.edu/~mathys/ecen1200/hwcl07/]

Les images vectorielles

L'image vectorielle est adaptée au travail sur des objets dont on connaît les paramètres de traçage ; elle est décrite en termes de formes élémentaires : lignes, cercles, rectangles, splines, courbes de Bézier. Les formes sont décrites par des attributs géométriques et par des attributs d'épaisseur, de couleur, de type. Une opération d'affichage ou d'impression nécessite une conversion en mode point (calculs) à cause des périphériques qui sont généralement en mode point. La taille d'une image vectorielle n'est fonction que de sa complexité.

Les avantages principaux de l'image vectorielle sont les suivants :

Indépendance vis à vis des périphériques : une image vectorielle est une image

symbolique interprétable (théoriquement) par chaque type de périphérique. Les modifications spatiales de l'image sont relativement souples car elles consistent en opérations géométriques ne conduisant pas à la perte d'information. L'image vectorielle est bien adaptée aux travaux de schématique : CAO, DAO. La taille d'une image vectorielle n'est fonction que de sa complexité; une image très complexe est de l'ordre de 1 à 2 Mo.

Les inconvénients principaux sont les suivants :

L'affichage implique des calculs car il faut transformer l'image vectorielle en une image matricielle. Un fichier vectoriel est en fait un programme; une dégradation d'une partie du fichier est désastreuse (un fichier en mode « texte « comme un fichier Postscript permet d'atténuer cette difficulté).

Codage d'images vectorielles

Le codage des images vectorielles peut se faire par une description dynamique à l'aide d'un programme de tracé .Elle peut s'exprimer à l'aide des codes de Freeman :sous sa forme la plus simple, le codage de Freeman décrit un pixel par rapport au pixel précédent par la direction du tracé. 8 directions de base sont définies.

Format d'images vectorielles

Nous présentons ici quelques formats d'images vectorielles dans le tableau ci-dessous:

Format	Description
PICT	déjà mentionné plus haut
DXF	origine Autocad, standard en CAO; adapté au travail en 2D; connu et très répandu
HPGL	origine Hewlett Packard; format répandu pour les périphériques detraçage

EPS	origine : format du langage Postscript pour Adobe; environnement PC et Mac
CGM	format dérivé de GKS; universel, normalisé
IGES	permet l'échange de données en CAO; environnement des stations de travail
EDIGEO	adapté à la cartographie; normalisé par l'AFNOR

Notons bien qu'il possible à l'aide des logiciels de traitement d'image de convertir une image vectorielle à une image matricielle. De plus pour le rendu d'une image vectorielle dans un écran d'ordinateur, une correspondance à une image matricielle est nécéssaire.

Exemple d'image vectorielle

Un exemple d'image vectorielle est également fourni par la figure ci-dessous.

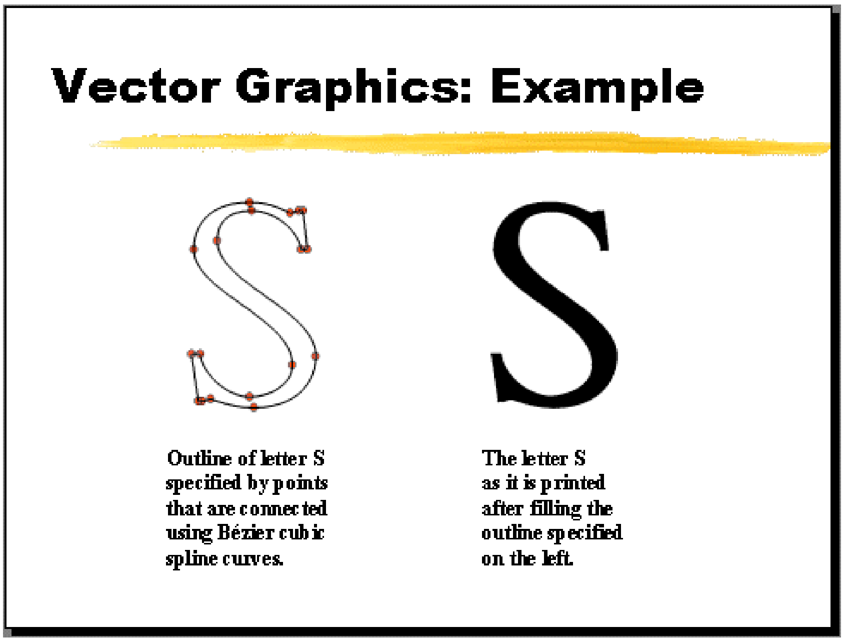


Figure 7: Une image vectorielle

[Source: http://ecee.colorado.edu/~mathys/ecen1200/hwcl07/]

La figure suivante montre une différence entre une image bitmap et une image vectorielle : l'image matricielle est un ensemble de point tandis que l'image vectorielle est définie à partir de l'equation d'un cercle.

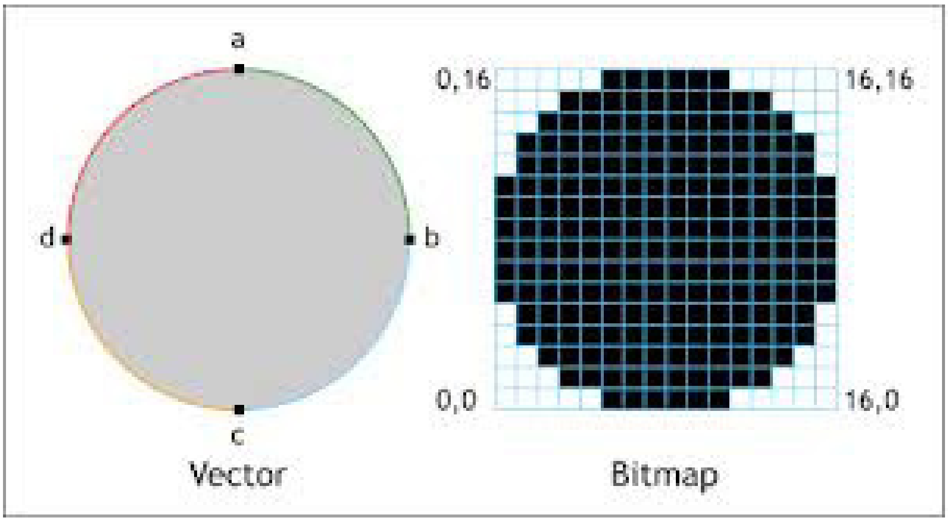


Figure 8 : lien entre une image vectorielle et une image bitmap

[Source: http://www.xaraxone.com/webxealot/xealot30/html/features.htm]

Techniques de pavage de l'espace et élements structurants d'une image

Dans une image matricielle, nous avons un ensemble de pixels en dimension 2, un ensemble de voxels en dimension trois ou un ensemble d' hypervoxels en dimension n. Un pixel est un carré. Un voxel est un cube. Un hypervoxel est un hypercube.

La figure 9 suivante montre respectivement en dimension 2 et 3, un pixel en (a) et un voxel en (b). Il en est de même de la figure 10 où nous avons des ensembles de pixels et de voxels.

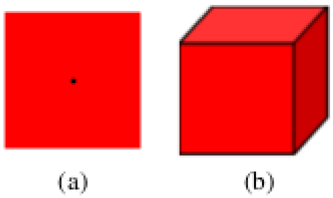


Figure 9 : un pixel en (a) et un voxel en (b)

Figure 10 : un ensemble de pixels et un ensemble de voxels

L'image peut se trouver dans un espace discret suivant une certaine dimension (plan et espace). Cela est illsustré par la figure ci-dessous où nous pouvons apercevoir des objets en dimension 2 et en dimension 3 réalisé avec le logiciel POV-RAY. Dans certaine domaine notamment en imagérie médicale, les images 3D sont utilisées précisément en Echographie.

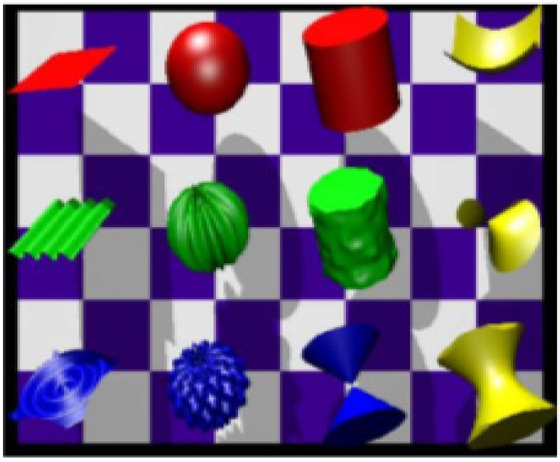


Figure 11 : plan et espace (une image générée avec POV-RAY) : des images en dimension 2 et en dimension 3

Il existe plusieurs techniques de découpage de l'espace en pavés : les diagrammes de Voronoï et de Delaunay et les transformations quasi-affines. Ces transformations permettent d'obtenir des pavés d'autres formes (rectangulaire, triangulaire) régulières ou irrégulières.

La figure 12 montre aussi un exemple de diagramme de Voronoï et de Delaunay permettant de créer une grille pour une image donnée.

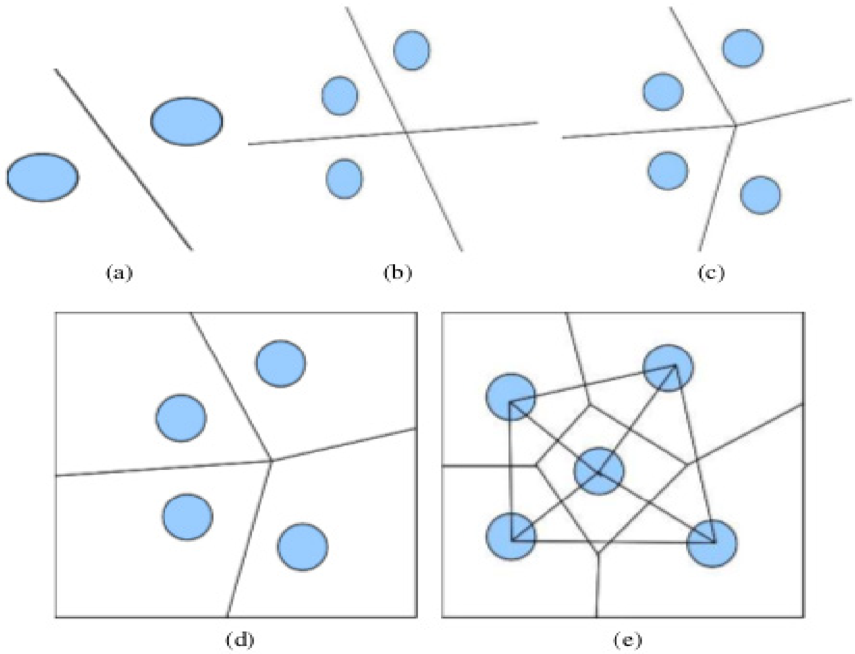


Figure 12 : diagramme de voronoï et de delaunay

Le pavage du plan par une transformation quasi-affine permet également de déterminer les élements structurants d'une image.

Une transformation quasi-affine de façon générale est définie par :

Définition *(Transformation Quasi Affine) On appelle Transformation Quasi Affine* T *une fonction définie par :*

$$T : \quad \mathbb{Z}^n \quad \rightarrow \quad \mathbb{Z}^n$$

$$(x_1, x_2, \ldots, x_{n-1}, x_n) \longmapsto \begin{cases} y_1 = & \left[\frac{a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n-1}x_{n-1} + a_{1n}x_n + b_1}{\omega}\right] \\ y_2 = & \left[\frac{a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n-1}x_{n-1} + a_{2n}x_n + b_2}{\omega}\right] \\ y_{n-1} = & \left[\frac{a_{n-11}x_1 + a_{n-12}x_2 + \ldots + a_{n-1n-1}x_{n-1} + a_{n-1n}x_n + b_{n-1}}{\omega}\right] \\ y_n = & \left[\frac{a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn-1}x_{n-1} + a_{nn}x_n + b_n}{\omega}\right] \end{cases}$$

où a_{ij} sont des entiers et ω un entier strictement positif. La transformation quasi affine est définie par sa matrice A=$\frac{1}{\omega}\left(a_{ij}\right)$ et son vecteur $\left(b_i\right)$.

Comme le montre la figure ci-dessous, nous obtenons des pavès de taille variée. Par l'intersection de droites paralèlles continues.

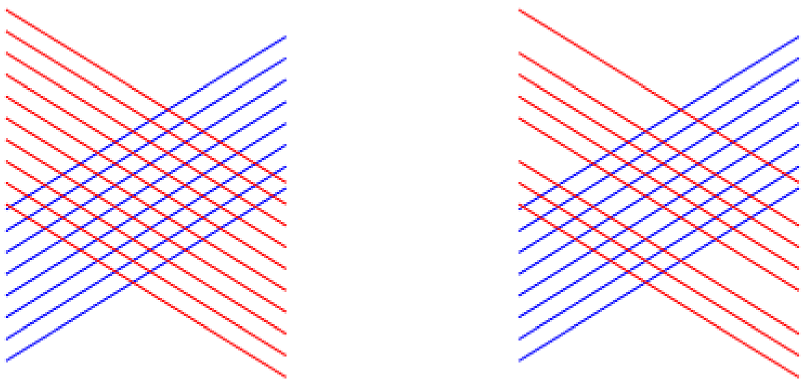


Figure 13 : un exemple de transformation quasi-affine en dimension 2

D'autres techniques de pavage existent comme les arbres kd-tree et peuvent avoir des applications en simulation. Un exemple de grilles est proposé par la figure suivante :

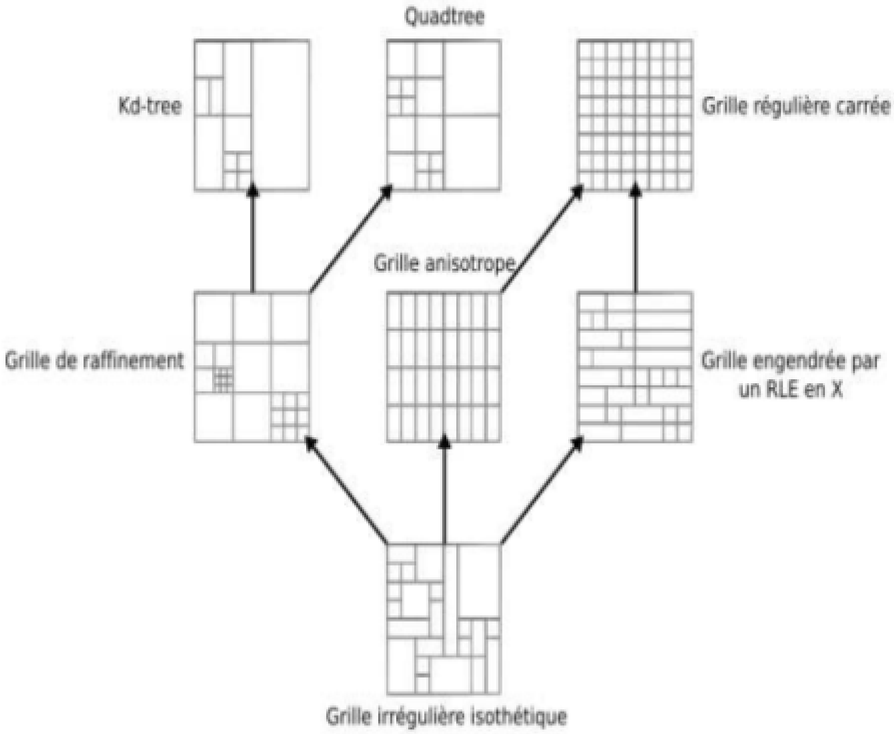


Figure 14 : autres pavages du plan

Conclusion

En somme, cette activité a permis de voir essentiellement les images vectorielles et matricielles avec leurs avantages et leurs inconvénients. Les modèles des élements structurants d'une image ont également été présentés avec des techniques de pavage de l'espace.

Évaluation

Exercice 1 : Proposer un algorithme qui permet de changer la couleurs d'une zone de pixels.

Écrire un algorithme qui génère un histogramme des couleurs de l'image initiale ;

Ecrire un programme qui convertit cette image en image noir et blanc.

Exercice 2 : quelle est la différence entre une image matricielle et une image vectorielle?

Activité 2 - Droite discrète de Bresenham et de Reveillès

Présentation

En infographie, dans la conception d'un schéma par exemple, nous avons recours dans certains cas de traçer un segment de droite. Le tracé d'un segment de droite suit un algorithme précis de tracé de droite discrète. Dans cette activité nous présenterons la droite discrète de Bresenham et la droite discrète de Reveillès.

Détails de l'activité

L'algorithme de Bresenham (1965)

L'algorithme de Bresenham permet de tracer une droite discrète. Pour deux points discrets A (x1 y1) et B (x2 y2) réliés par une droite continue, la droite discrète de Bresenham est constituée des points discrets proches de la droite continue : La figure ci-dessous donne un exemple de droite discrète de Bresenham où nous voyons les points discrets proches de la droite continue

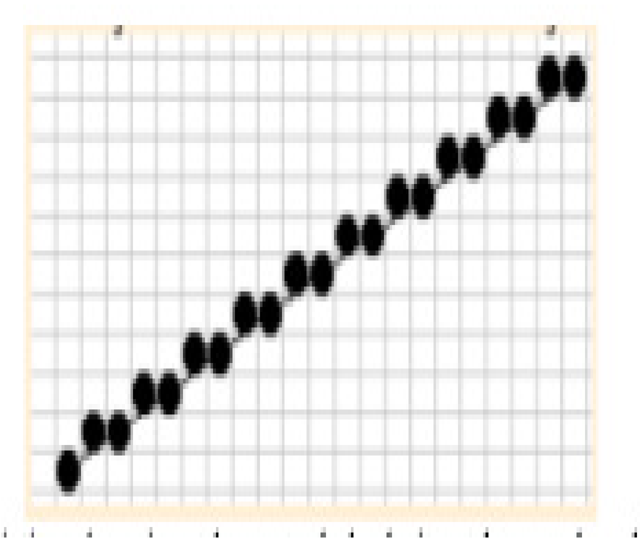
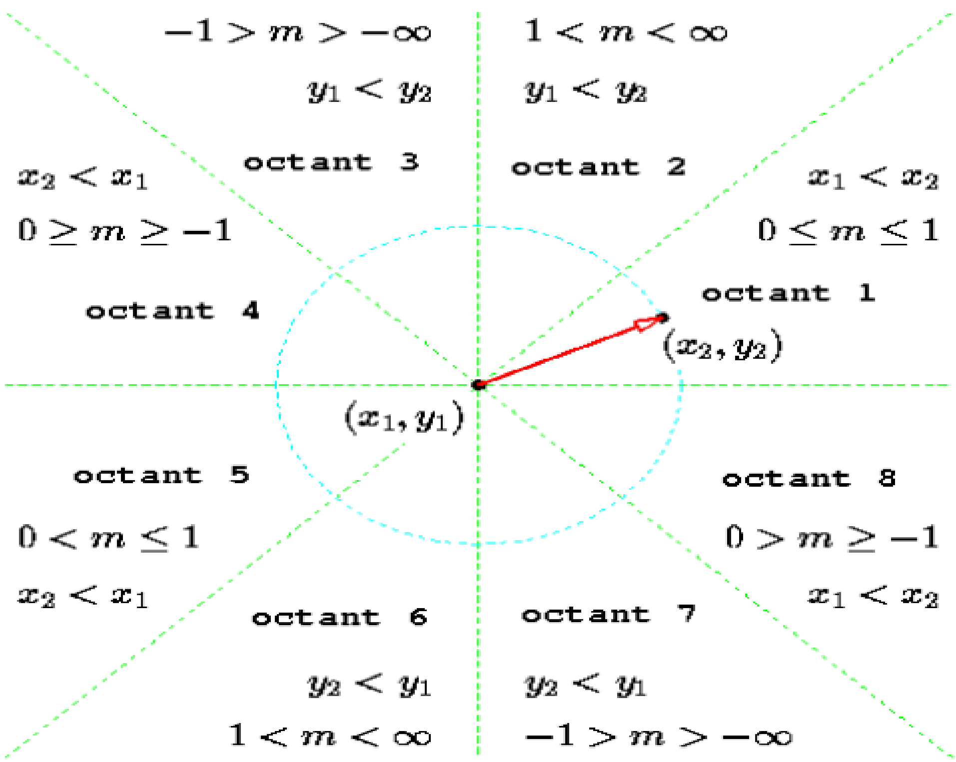
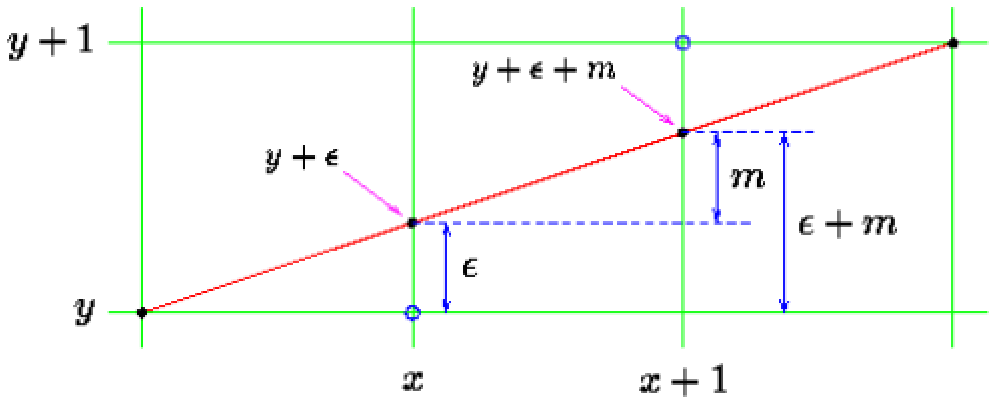


Figure 1 : une droite discrète de Bresenham

Pour le tracé d'une droite discrète de Bresenham, nous avons 8 cas c'est à dire 8 octants où se fera l'analyse des points discrets :



Nous avons donc dans l'octant 1 par exemple la configuration suivante :



Algorithme de tracé de droite discrète de Bresenham dans le premier octant

$\varepsilon \leftarrow 0$; $y \leftarrow y1$; $x \leftarrow x1$;
TantQue $x \leq x2$ faire

dessiner (x,y)
Si ($\varepsilon+m < 0.5$) alors
$\varepsilon \leftarrow \varepsilon + m$
Sinon
$y \leftarrow y+1$; $\varepsilon \leftarrow \varepsilon + m - 1$
FinSi
$x \leftarrow x+1$

FinTantQue

En exercice, nous allons analyser l'algorithme pour les autres octants.

La droite discrète analytique de Reveillès (1989)

La droite discrète analytique de Reveillès a été proposée par Jean-Pierre Reveillès.

Définition ***(Droite analytique discrète 2D)***

La droite discrète $D(a,b,\mu,\omega)$ est l'ensemble des points $(x, y) \in \mathbb{Z}^2$ tels que $\mu \leq ax+by < \mu + \omega$ avec $\mu \in \mathbb{Z}$ $\omega \in \mathbb{Z}$, $(a,b,) \in \mathbb{Z}^3$ et $PGCD(a,b)=1$.

A partir de cette définition, la droite discrète analytique de Reveillès est un ensemble de points discrets situés entre deux droites continues, paralèlles.

Plusieurs variantes de droites discrètes telles que mince, standard, naîf, épaisse et super-couverture existent. La figure suivante donne des exemples de ces droites discrètes analytiques.

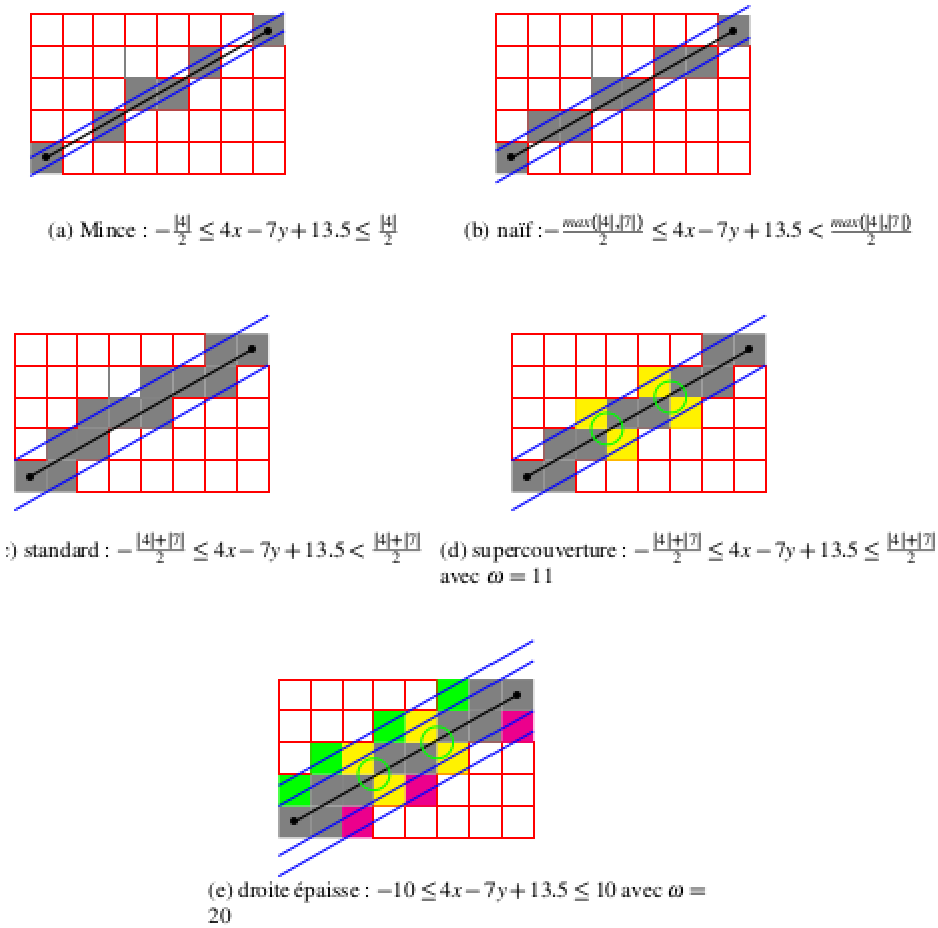


Figure 2 : droites discrètes analytiques de Reveillès

[Source: Abdoulaye SERE, transformations analytiques appliquées aux images multi-échelles et bruitées, thèse en informatique, 2013]

Et plus généralement, en dimension 3, nous obtenons le plan standard et naif comme le montre la figure ci-dessous.

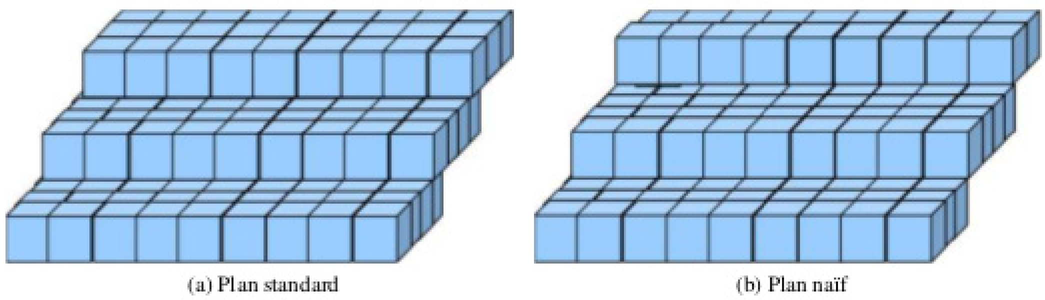


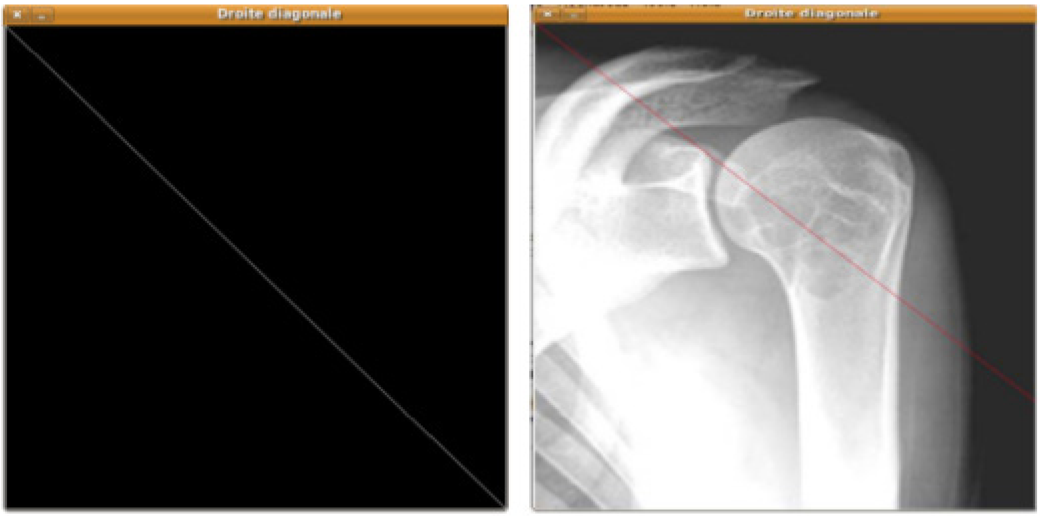
Figure 3 : deux plans discrets

[Source: Abdoulaye SERE, transformations analytiques appliquées aux images multi-échelles et bruitées, thèse en informatique, 2013]

La droite analytique discrète est un cas particulier de l'hyperplan analytique discret défini par

Définition (Hyperplan analytique discret) *En dimension n, l'hyperplan analytique discret de paramètres* $A(A_1, A_2,\ldots,A_{n-1}, A_n) \in \mathbb{R}^n$, $\mu \in \mathbb{R}$ *et* $\omega \in \mathbb{R}$ *est l'ensemble des points* $X = (x_1, \ldots, x_n) \in \mathbb{Z}^n$ *vérifiant* $\mu \leq \sum_{i=1}^{n}(A_i x_i) < \mu + \omega$

Voici deux exemples de droites discrètes naïves tracées à l'aide de la librairie Opencv sur une image existante en (a) et en (b) :



(a) (b)

Figure 4 : Droites discrètes naïves en (a) et en (b)

[Source: Abdoulaye SERE, transformations analytiques appliquées aux images multi-échelles et bruitées, thèse en informatique, 2013]

Conclusion

Dans cette activité, nous avons analysé la droite discrète de Bresenham et la droite discrète de Reveillès. La droite discrète de Reveillès est un cas particulier en dimension de 2 de la définition généraliste de l'hyperplan analytique discret en dimension n.

Évaluation

Exercice 1 : Proposer un algorithme de tracé d'une droite discrète de Bresenham à partir de deux points discrets quelconques.

Exercice 2 : Proposer un algorithme de tracé d'une droite discrète analytique à partir de deux points discrets quelconques.

Exercice 3 : soient A (5, 6) et B (10, 7). Proposer la définition mathématique du segment de droite standard [AB].

Activité 3 - Cercle discret, courbes discrétes quelconques

Introduction

Dans cette activité, nous étudierons les polygones, le cercle analytique discret et le cercle discret de Bresenham. Une définition des courbes discrètes quelconques sera également proposée.

Détails de l'activité

Polygones

En infographie, dans la conception d'un dessin, les polygones sont souvent utilisés. Ils sont construits sur la base   de segments de droites comme la montre la figure.

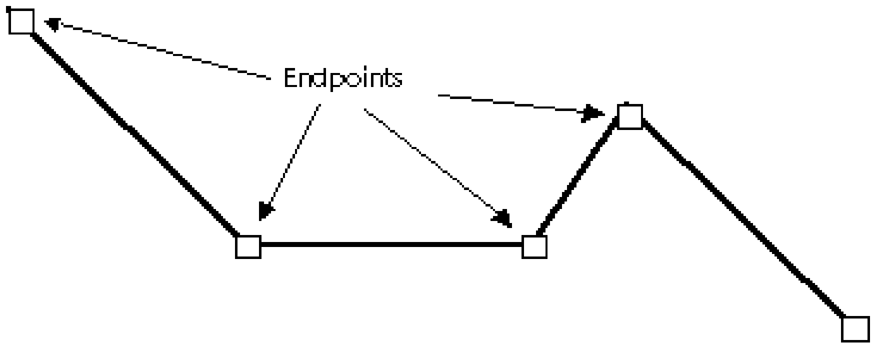


Figure 1   : un ensemble de segments de droites flexibles

[Source: http://www.webopedia.com/TERM/P/polyline.html]

Dans certains cas, nous avons des polygones convexes ou concaves comme le montre la figure suivante.

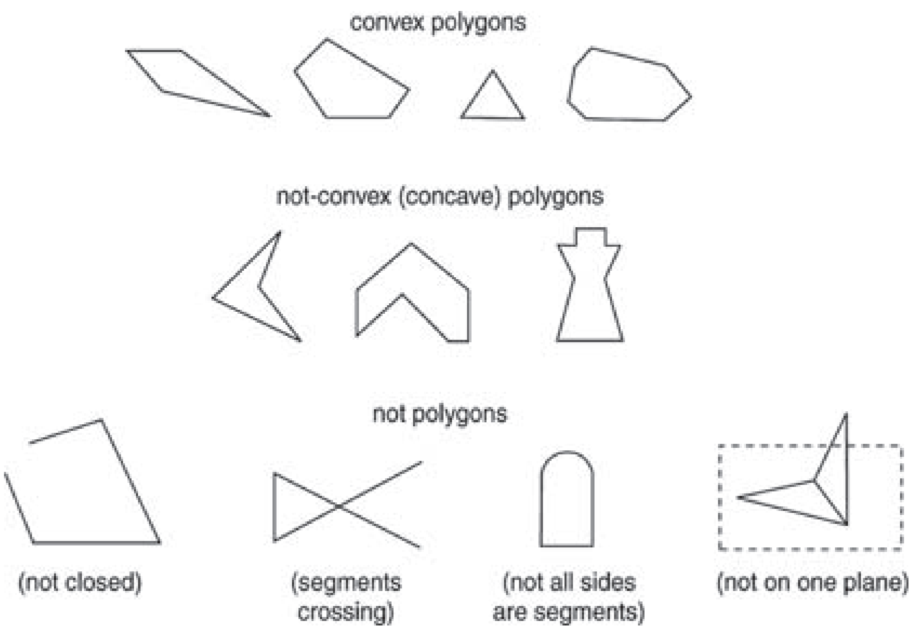


Figure 2 : Polygones

[Source:http://www.cliffsnotes.com/math/geometry/polygons/classifying-polygons]

En infographie, les polygones avec des couleurs de remplissage peuvent aussi être utilisés pour traiter les informations ou inclure des textures dans les polygones.

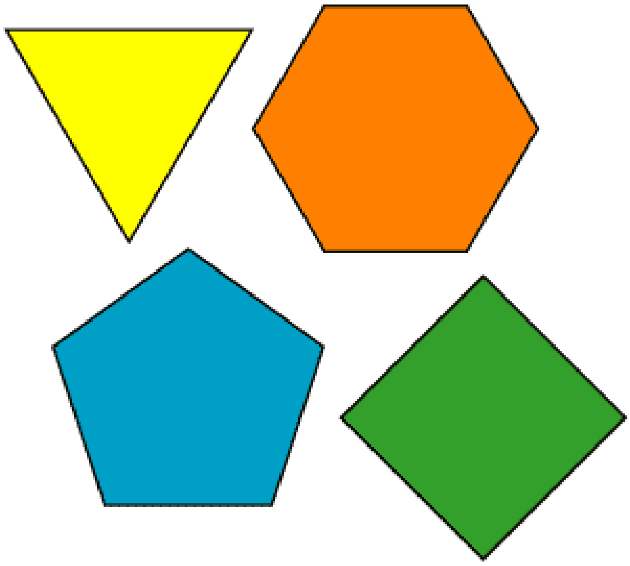


Figure 3 : Polygones remplies par une couleur

[Source: http://www.barryscientific.com/lessons/polygon.html]

Cercle discret de Brésenham

Il existe également un algorithme de Bresenham pour le tracé de cercles discrets. C'est l'ensemble des points discrets proches du cercle continu. Dans la figure suivante, nous observons un arc de cercle où les points discrets retenus sont ceux qui sont proche de l'arc de cercle continu.

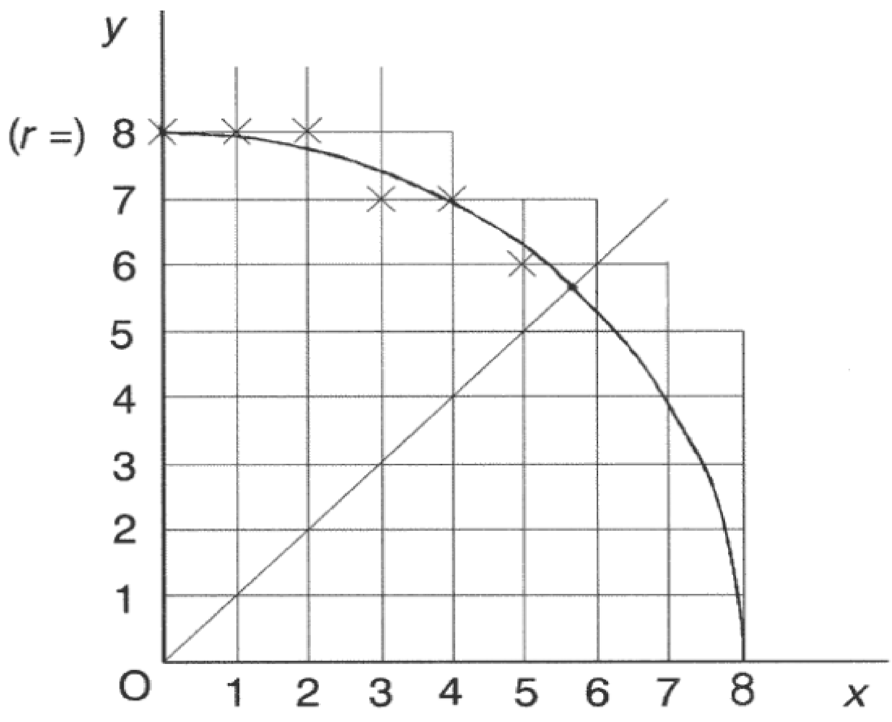


Figure 4 : cercle discret de Bresenham

L'algorithme de Michener est une extension de l'algorithme de Bresenham pour prendre en compte le cercle discret.

```
algorithme de Michener pour l'octant 1 et
pour un cercle de centre (0,0) et de rayon R
début
        // Initialisation
        x = R ; y = 0 ;
        TantQue x>= y faire
             Dessiner(x,y)
             y = y + 1 ;
             d1 = |R*R-(x*x+y*y)| ;
             d2 = |R*R-((x-1)*(x-1)+y*y)| ;
             Si d1 > d2 alors
                   x = x - 1 ;
             FinSi
      FinTantQue
fin
```

En exercice, nous allons analyser l'algorithme de Bresenham pour les autres octants.

Des exemples de cercles discrets de Bresenham sont fournis par la figure suivante.

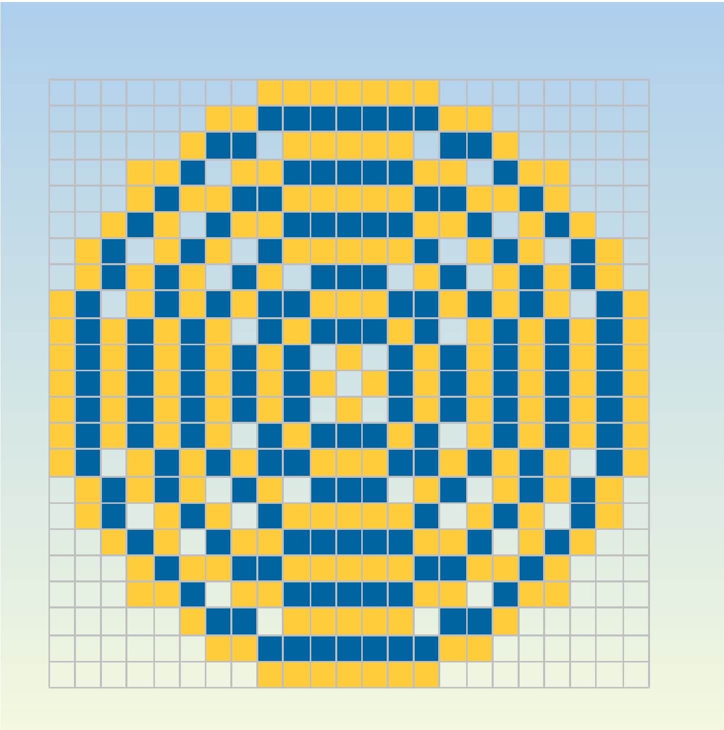


Figure 5 : Cercles discrets de Bresenham

Une autre definition du cercle discret est le cercle discret analytique.

Cercle discret analytique

Comme la droite discrète de Reveillès, le cercle discret analytique est aussi un cas particulier de la définition de l'hypersphère analytique discrète. L'hypersphère analytique discrète est définie par :

Définition ***(Hypersphère arithmétique analytique discrète***) *Soit d la dimension l'espace. Soit $r \in R_+^*$, et $o=(o_1, o_2, \ldots, o_d) \in \mathbb{R}^d$. Soit $\omega_1 : \mathbb{R}^d \longrightarrow \mathbb{R}_-$ et $\omega_2 : \mathbb{R}^d \longrightarrow \mathbb{R}_+$ sont des fonctions changement de variables. L'hypersphère arithmétique analytique discrète $S(o, r, \omega_1, \omega_2)$ de centre o et rayon r et de fonction d'épaisseur ω_1, ω_2, est le sous ensemble de points $v \in \mathbb{Z}^d$ défini par :*

$$S(o, r, \omega_1, \omega_2) = \{v \in \mathbb{Z}^d \setminus \omega_1(v) \leq \sum_{i=1}^{d} (v_i - o_i)^2 - r^2 < \omega_2(v)\}$$

Nous obtenons de cette définition l'algorithme suivant :

Algorithme 1.3 Tracé d'une hypersphère analytique discrète

DONNÉES :
- Soit S un ensemble de m pavés pixels $P_1, \ldots, P_m$ servant à représenter l'espace image pour le tracé de l'hyspershère analytique en dimension n. Un pavé P_i a pour centre $(x_1^{P_i}, x_2^{P_i}, \ldots, x_{n-1}^{P_i}, x_n^{P_i})$,
- Soient n la dimension de l'espace et $r \in R_+^*$, $o=(o_1, o_2, \ldots, o_d) \in \mathbb{R}^d$ et $\omega \in R_+^*$.

DÉBUT

Pour chaque pavé P_i de S.
  Calculer ValeurSomme=$\sum_{i=1}^{n} (x_i^{P_i} - o_i)^2$
  Si $((r - \frac{\omega}{2})^2 \leq$ValeurSomme) et (ValeurSomme$<(r + \frac{\omega}{2})^2$) alors
  Allumer P_i comme étant un pavé d'une hypersphère analytique
  Finsi
FinPour

FIN

En dimension 2, le cercle discret analytique est déduit de cette définition générale d'une hypersphère analytique :

Cercle discret analytique

Le cercle discret analytique $\mathbb{C}(\mathrm{M}_0, r, \omega)$ de centre $\mathrm{M}_0 = (x_0, y_0) \in \mathbb{R}^2$, de rayon $r \in \mathbb{R}_+^*$ et d'épaisseur $\omega \in \mathbb{R}_+^*$, est l'ensemble suivant de $\mathbb{Z}^2$:

$$\mathbb{C}(\mathrm{M}_0, r, \omega) = \left\{ (i,j) \in \mathbb{Z}^2 | \left(r - \frac{\omega}{2}\right)^2 \leq (i - x_0)^2 + (j - y_0)^2 < \left(r + \frac{\omega}{2}\right)^2 \right\}.$$

La figure ci-dessous montre ainsi des exemples de cercles discrets analytiques.

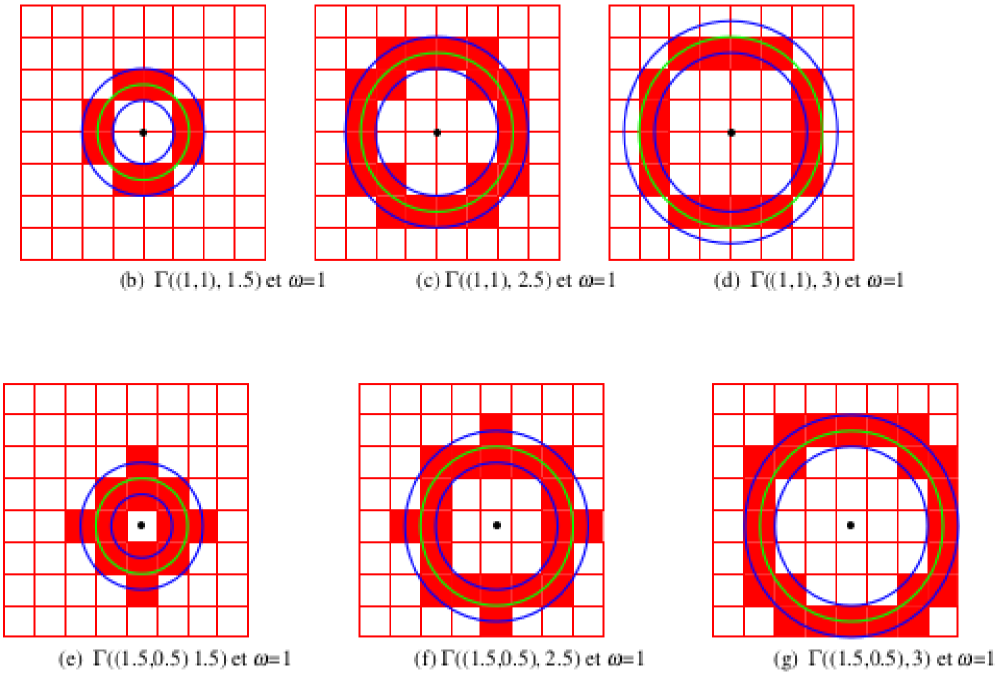


Figure 6 : des exemples de cercles analytiques discrets

[Source: Abdoulaye SERE, transformations analytiques appliquées aux images multi-échelles et bruitées, thèse en informatique, Université de Ouagadougou, 2013]

Nous avons ici un exemple de cercle discret tracè à l'aide de la librairie Opencv

Figure 7 : un exemple de cercles analytiques discrets en rouge dans un espace discret

[Source: Abdoulaye SERE, transformations analytiques appliquées aux images multi-échelles et bruitées, thèse en informatique, Université de Ouagadougou 2013]

De façon, générale comment pouvons tracer une courbe quelconque? L'idée est de pouvoir tracer des courbes discrètes dans un espace discret en suivant une correspondance en le monde continu et le monde discret comme le montre la figure ci-dessous :

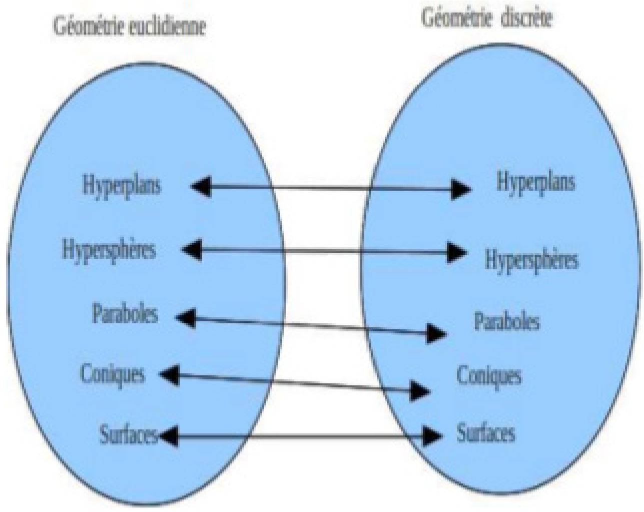


Figure 8 : liens entre des courbes continues et des courbes discrètes

[Source: Abdoulaye SERE, transformations analytiques appliquées aux images multi-échelles et bruitées, thèse en informatique, Université de Ouagadougou 2013]

Discrétisation d'une courbe ou d'une surface

La discrétisation consiste à obtenir un ou plusieurs objets discrets à partir d'un objet continu.

Plusieurs techniques de discrétisation existent pour un contour continu : la discrétisation au plus proche, la discrétisation interne, la discrétisation extérieure. Ces méthodes permettent de discrétiser des contours fermés continus soit par des points discrets internes ou externes ou en ne s'intéressant qu'aux points discrets proches du contour continu. Les trois variantes de discrétisation sont présentées en (a), en (b) et en (c) dans la figure ci-dessous.

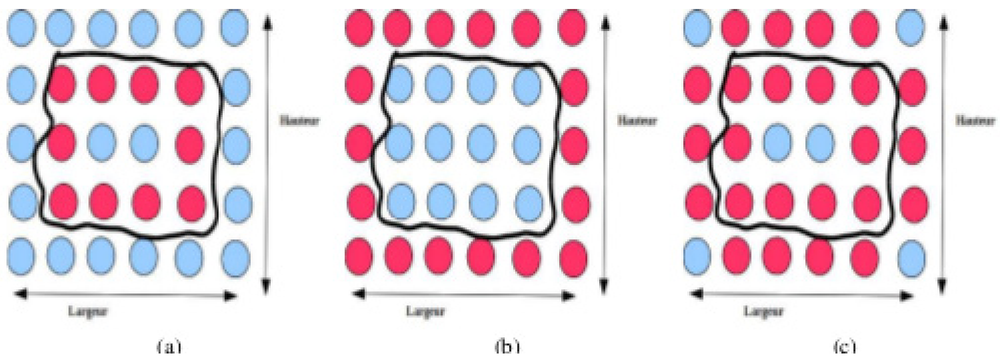


Figure 9 : discrétisation d'une courbe continue quelconque

[Source: Abdoulaye SERE, transformations analytiques appliquées aux images multi-échelles et bruitées, thèse en informatique, Université de Ouagadougou, 2013]

La discrétisation discrète analytique peut être définie par :

Définition *(Discrétisation d'un objet continu). Soit un E objet continu en dimension n et d une distance, la discrétisation de E notée $D_d(E)$ associée à la distance d est définie par : $D_d(E)=\{p\in\mathbb{Z}^n \mid d(p, E) \leq \frac{\alpha}{2}\}$ avec $\alpha\in\mathbb{N}$-{0}.*

Cette définition prend en compte la discrétisation au plus proche, la discrétisation interne, la discrétisation extérieure dans la mesure où les points discrets sont situés par rapport au contour continu en prenant en compte la notion de distance.

Conclusion

Dans cette activité, nous avons présenté les courbes discrètes allant des cercles discrets et des courbes discrètes quelconques.

Évaluation

Exercice 1 :proposer un exemple de cercle discret analytique et proposer un algorithme de tracé de ce cercle analytique.

Exercice 2 : proposer l'algorithme de Bresenham dans le cas du cercle discret pour chaque octant.

Résumé de l'unité

Les primitives en géométrie discrète sont la droite discrète de Bresenham et de Reveillès, le cercle discret, et plus générallement les hyperplans discrets analytiques, les hypersphères discrète analytique et les courbes discrètes analytiques quelconques.

Évaluation de l'unité

Vérifiez votre compréhension!

Lectures et autres ressources

Les lectures et autres ressources de cette unité se trouvent au niveau des lectures et autres ressources du cours.

Unité 2: Segmentation, Filtrage et Morphologie des Images

Introduction à l'unité

Dans cette unité, nous évoquerons les techniques de segmentation d'une image, certains filtres de traitement d'images et des éléments de morphologie d'image (. Ces opérations permettent d'extraire des composants d'une image. Elles servent également dans la réalisation d'effets spéciaux.

Nous divisons l'unité en trois activités qui sont la segmentation, le filtrage et la morphologie.

Objectifs de l'unité

À la fin de cette unité, vous devriez être capable de:

1. Filtrer une image pour réduire le bruit présent ;
2. Détecter les contours d'une image en application des filtres;
3. Transformer une image par dilatation ou par érosion pour construire des effets spéciaux ;
4. Extraire des éléments d'une image par segmentation.

Termes clés

Segmentation : la segmentation consiste à déterminer les différentes régions d'une image

Filtrage: Le filtrage est une opération de combinaison linéaire (ou non) de pixels de I, produisant une image R

Activités d'apprentissage

Activité 1 - Segmentation

Introduction

La segmentation d'image est une opération de traitement d'images qui consiste à rassembler des pixels ayant des caractéristiques communes. C'est une partition de l'ensemble des pixels de l'image en différents groupes. Chaque groupe est supposé correspondre à un objet de l'image. Les pixels sont ainsi regroupés en régions, qui constituent une partition de l'image.

Dans cette activité, nous presenterons quelques méthodes de segmentation (ils existent plusieurs méthodes). D'autres éléments de la segmentation seront révus dans les activités suivantes.

Détails de l'activité

Ils existent trois approches de segmentation :

1. La segmentation par approche région ;
2. La segmentation par approche frontière ou contour ;
3. La segmentation par seuillage des pixels en fonction de leur intensité.
4. La segmentation par approche région
5. La technique de l'histogramme des couleurs d'une image permet de détecter des régions : pour une image donnée on pourra représenter l'histogramme des couleurs (rouge, verte et bleue) comme le montre la figure ci-dessous

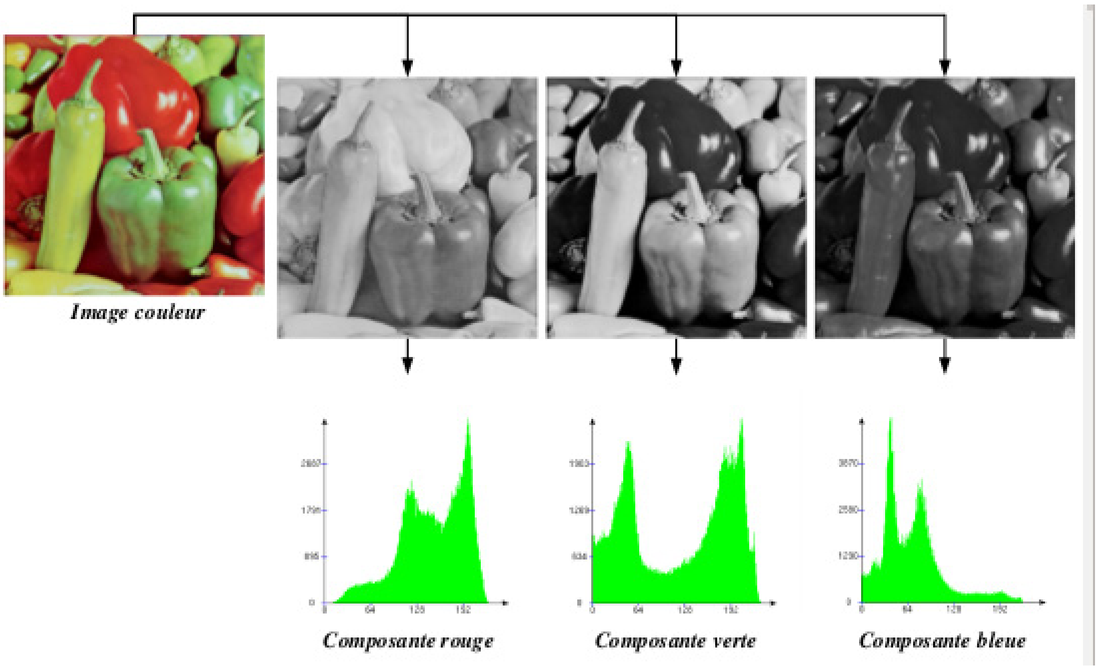


Figure 1 : Histogramme des couleurs d'une image

Une application récursive de l'histogramme des couleurs permet de déterminer les région comme le montre la figure ci-dessous.

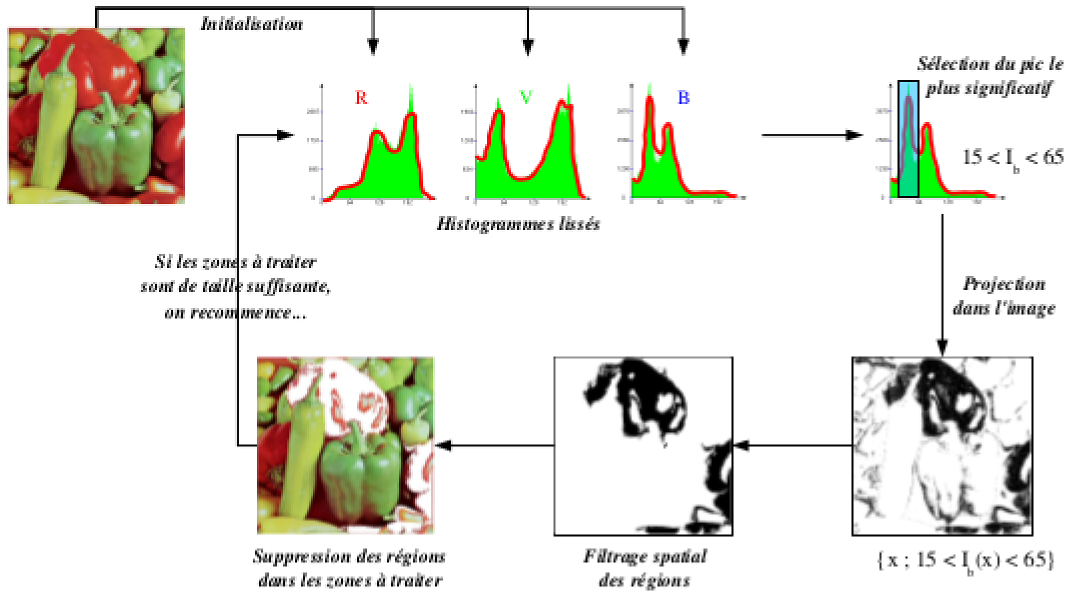


Figure 2 : une application récursive de l'histogramme des couleurs pour déterminer les régions d'une image

Il existe d'autres techniques de segmentation par région comme la méthode du split and merge.

La segmentation par approche frontière ou contour

Cette technique consiste à déterminer les contours de l'image pour en déterminer les différentes régions. L'activité 2 présente plusieurs filtres permettant de mettre en relief les contours d'une image.

La segmentation par seuillage

Il s'agit de faire une segmentation de l'histogramme des niveaux de gris de l'image. Cela qui donne une segmentation de l'image. Le problème se ramène à trouver le seuil de niveau de gris T optimal. L'image segmentée est alors une image binaire définie par J(x, y) =1 si (I (x, y)≥T) et J(x, y) =0, sinon . Les figures suivantes présentent des exemples de seuillage avec T variant :

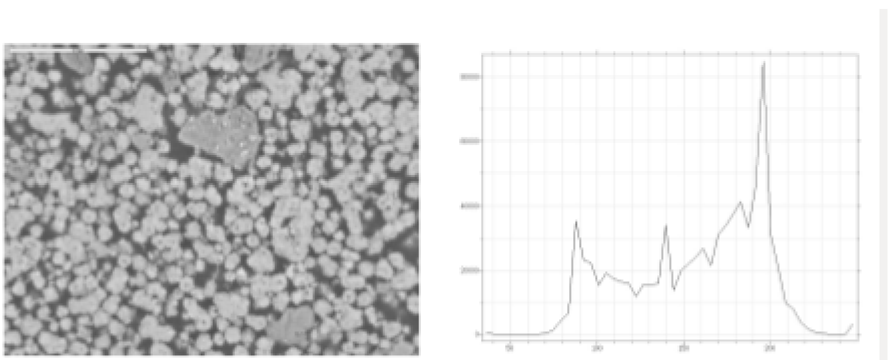
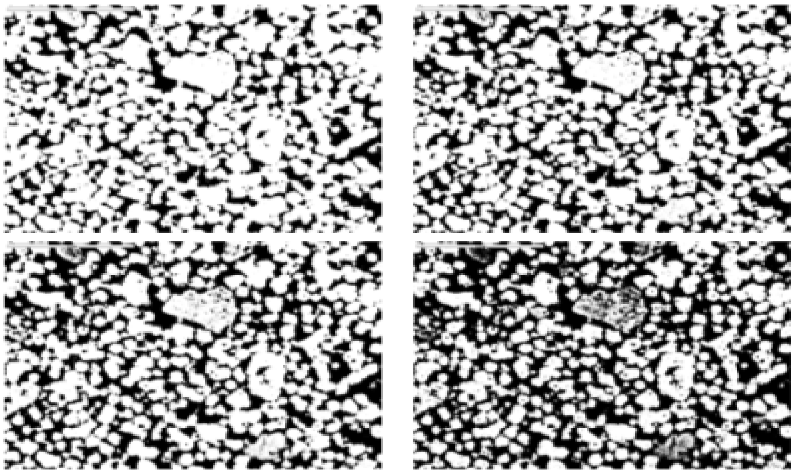


Figure 3 : à gauche, une image originale, à droite, un histogramme à niveaux de gris correspondant à cette image

La binarisation de cette image à des seuils T variant donne les images suivantes :



Binarisation avec différents seuils:135, 145, 155 et 165.

Figure 4 : Binarisation avec différents seuils

Conclusion

Dans cette activité, nous avons présenté brièvement la définition de la segmentation et les différents types de segmentation.

Évaluation

Exercice 1 : proposer un algorithme de segmentation par seuillage d'une image avec la valeur passée en paramètre.

Exercice 2: proposer un algorithme qui génère l'histogramme des couleurs d'une image donnée

Activité 2 -Filtrage d'une image

Présentation

Dans cette unité, nous présentons brièvement quelques filtres permettant de détecter des contours ou de réduire le bruit présent dans une image. Nous nous intéressons particulièrement aux filtres spatiaux linéaires et non linéaires.

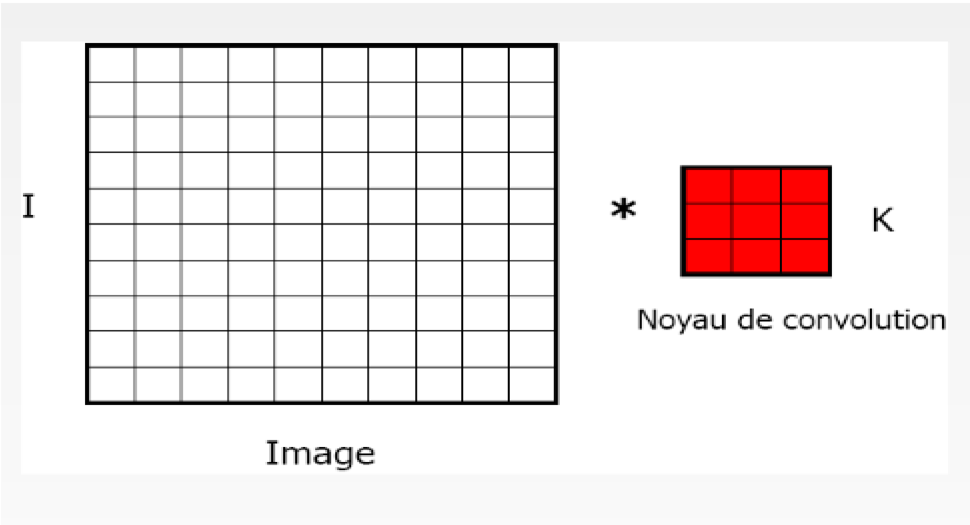
Détails de l'activité

Le filtrage spatial 2D est une convolution entre une image I(x, y) et un filtre K, appelé masque de convolution. Le filtrage est une opération de combinaison linéaire (ou non) de pixels de I, produisant une image R. L'opérateur h définit sur chaque pixel (x, y) de I et son voisinage, sa valeur de combinaison.

Le produit de convolution d'un signal 2D, I(x, y) (une image) avec un filtre k(i, j) est donné par :

$$R(x,y) = \sum_{u=-n}^{u=n} \sum_{v=-n}^{v=n} I(x+u, y+v).K(u+n, v+n)$$

En général, K est un masque carré de taille d impaire. Cette formule donne l'image filtrée R(x, y) est représentée graphiquement par :



Ainsi, nous voyons dans cette formule que le centre du filtre est superposé avec le pixel de l'image initiale. Beaucoup de traitements d'images sont basés sur les produits de convolutions.

Cette convolution peut être représentée par deux tableaux :

I(1,1)	I(1,2)	.	.					
I(2,1)	.							
.		.						
.			I(x,y)					
								I(n,m)

*

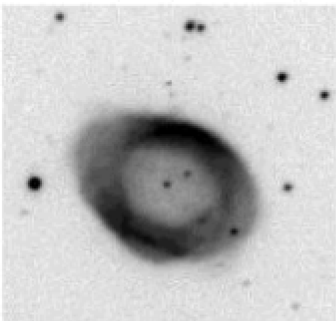
K(1,1)	K(2,1)	K(3,1)
K(2,1)	K(2,2)	K(2,3)
K(3,1)	K(3,2)	K(3,3)

Ainsi un exemple de cette convolution pour mettre en évidence l'apparition d'une variation d'intensité dans une image est illustrée par les tableaux suivants :

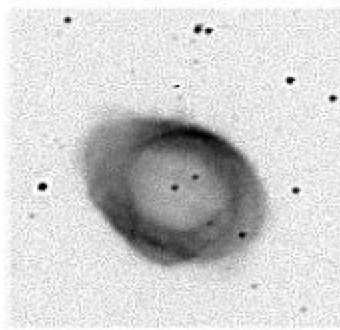
$$\begin{bmatrix} 100 & 100 & 100 & 100 & 100 \\ 100 & 100 & 100 & 100 & 100 \\ 100 & 100 & 150 & 100 & 100 \\ 100 & 100 & 100 & 100 & 100 \\ 100 & 100 & 100 & 100 & 100 \end{bmatrix} * \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 100 & 100 & 100 & 100 & 100 \\ 100 & 100 & 50 & 100 & 100 \\ 100 & 50 & 350 & 50 & 100 \\ 100 & 100 & 50 & 100 & 100 \\ 100 & 100 & 100 & 100 & 100 \end{bmatrix}$$

Le filtre "passe-haut"

Un filtre « passe haut » amplifie les hautes fréquences spatiales (les contours), comme les détails, et de ce fait, il améliore le contraste. Un filtre « passe haut » est caractérisé par un noyau comportant des valeurs négatives autour du pixel central, comme dans l'exemple ci-dessous:



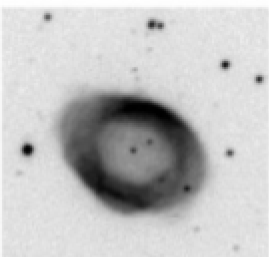
$$* \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} =$$



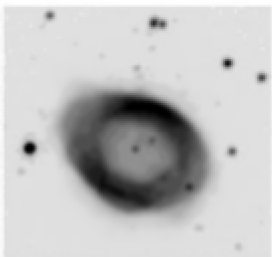
Par contre à cause de la valeur 5, élévé, dans une zone de pixels uniformes, ce filtre crée du bruit.

Le filtre "passe-bas"

Les filtres « passe bas » agissent en sens inverse des filtres « passe haut » et le résultat est un adoucissement des détails ainsi qu'une réduction du bruit granuleux.

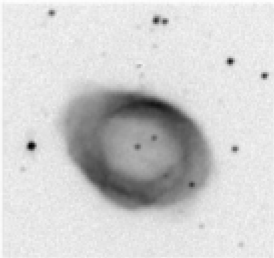


$$* \begin{bmatrix} 1 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 1 \end{bmatrix} =$$

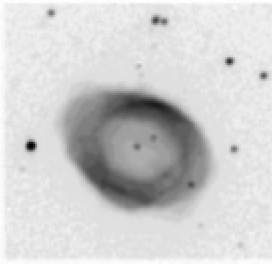


Le filtre moyen

C'est un cas particulier de filtre de convolution « passe-bas », qui remplace chaque pixel par la moyenne des valeurs des pixels adjacents et du pixel central.



$$* \begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix} =$$



Le filtre Laplacien

Le filtre Laplacien est un filtre de convolution particulier utilisé pour mettre

en valeur les détails qui ont une variation rapide de luminosité. Le Laplacien est donc idéal pour rendre visible les contours des objets, d'où son utilisation dans la reconnaissance de formes dans des applications militaires, puis civiles.

D'un point de vue mathématique, le Laplacien est une dérivée d'ordre 2, à

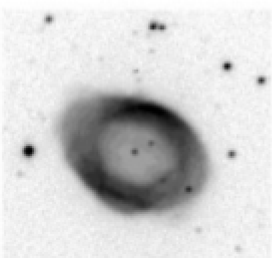
deux dimensions, en formule cela donne:

$$L(x,y) = \frac{\partial^2 I(x,y)}{\partial x^2} + \frac{\partial^2 I(x,y)}{\partial y^2}$$

On obtient dans l'espace discret trois filtres :

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

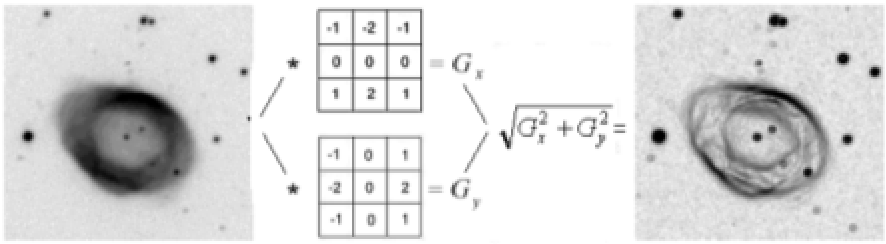
Un exemple d'application du filtre Laplacien est proposé par :



$$* \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} =$$

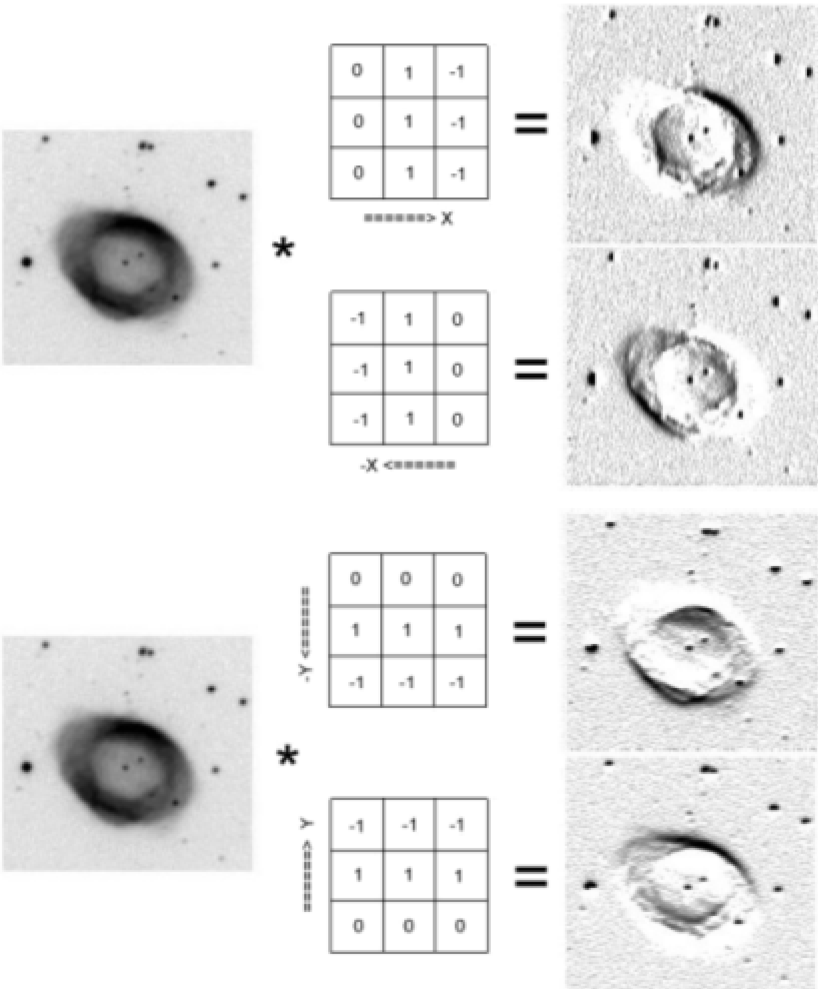
Le filtre Sobel

Le filtre de Sobel utilise par exemple deux noyaux 3x3, l'un pour l'axe horizontal (X) et l'autre pour l'axe vertical (Y). Chacun des noyaux est en fait un filtre gradient, qui sont tous les deux combinés pour créer l'image finale. Ce filtre est illustré par :

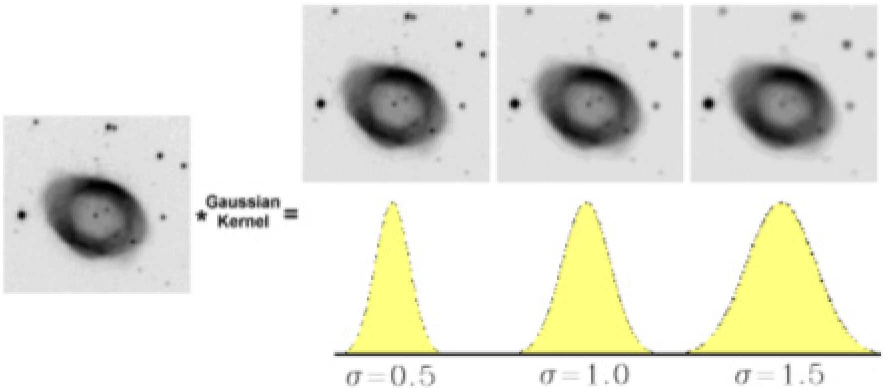


Le filtre médiane ne réalise pas une convolution mais utilise un noyau pour collecter les données nécessaires afin de trouver la valeur médiane d'une liste 3X3, à retenir pour le pixel de l'image filtrée.

Le filtre gradient : un exemple de filtre gradient est proposé par :



Le filtre Gaussien utilise une fonction gaussienne classique : Dans le traitement d'images on traite des données à deux dimensions (X et Y), on introduit donc une fonction gaussienne à deux dimensions G(x, y). Un exemple est proposé par :



Ils existent d'autres filtres CANNY, PREWITT, FREEMAN, ET KIRSCH.

Dans certains logiciel comme GIMP, il est possible d'appliquer certains filtres comme le Laplacien, le Gradient, le filtre de SOBEL ou de définir son propre filtre : GIMP utilise des matrices de convolution 5x5 ou 3x3. (avec GIMP 2.8.10 en allant dans Filtres ⍰ Génériques ⍰ Matrice de Convolution). Le logiciel Astroart permet également d'appliquer des filtres sur une image.

Conclusion

Dans cette activité, nous avons présenté la définition du filtrage et quelques filtres, permettant de réduire le bruit présent dans une image ou de déterminer ses contours.

Évaluation

Exercice 1 : proposer un algorithme pour le filtre mediane et Gradient, Gaussien

Exercice 2 : proposer un algorithme pour le filtre moyenne, le filtre "passe haut" et "passe bas"

Activité 3 - Morphologie

Introduction

La morphologie mathématique est une technique mathématique et informatique d'analyse de structures qui est liée à l'algèbre, la théorie des treillis, la topologie et les probabilités

Dans cette activité, nous étudierons les éléments de la morphologie suivants :

La dilatation

L'érosion

L' ouverture

La fermeture

La squelettisation

Détails de l'activité

Les opérations morphologiques sont des filtres non-linéaires qui peuvent s'appliquer tant aux images binaires qu'à celles à niveaux de gris.

Quatre (4) opérations morphologiques de base:

La dilatation

L'érosion

L'ouverture

La fermeture

La morphologie mathématique répose sur la notion d'élément structurant. Un élément structurant est composé :

D'un pixel central (en noir)

D'un ensemble de pixels (en gris)

La figure suivante donne un exemple d'éléments structurants

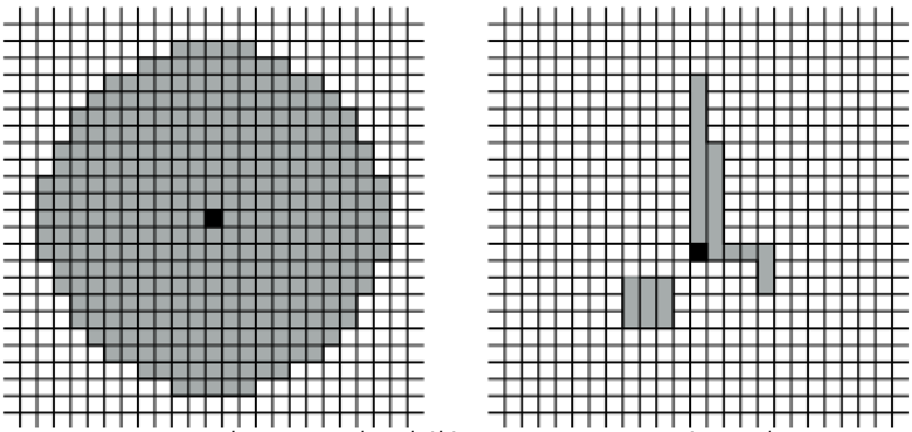


Figure 1 : des exemples d'éléments structurants à gauche (connexes) et à droite (disjoints)

Dilatation

L'effet de la dilatation est d'abord d'élargir la figure. La hauteur et largeur de la figure dilatée seront les sommes respectivement des hauteurs et largeurs de la figure originelle et de l'élément structurant. Si l'élément structurant est décentré, la dilatation décalera la figure dans le même sens. Enfin les coins convexes de la figure seront déformés en fonction de l'élément structurant.

Soit X une figure, à savoir un ensemble de pixels. Pour un élément structurant B, la dilatation de X par B est l'ensemble obtenu en remplaçant chaque pixel p de X par sa fenêtre Bp :

$$\mathrm{Dil}_B(X) = \cup \{B_p \mid p \in X\}.$$

Bp est la fenêtre obtenue en positionnant B sur p. La figure suivante illustre un exemple de dilatation d'un objet.

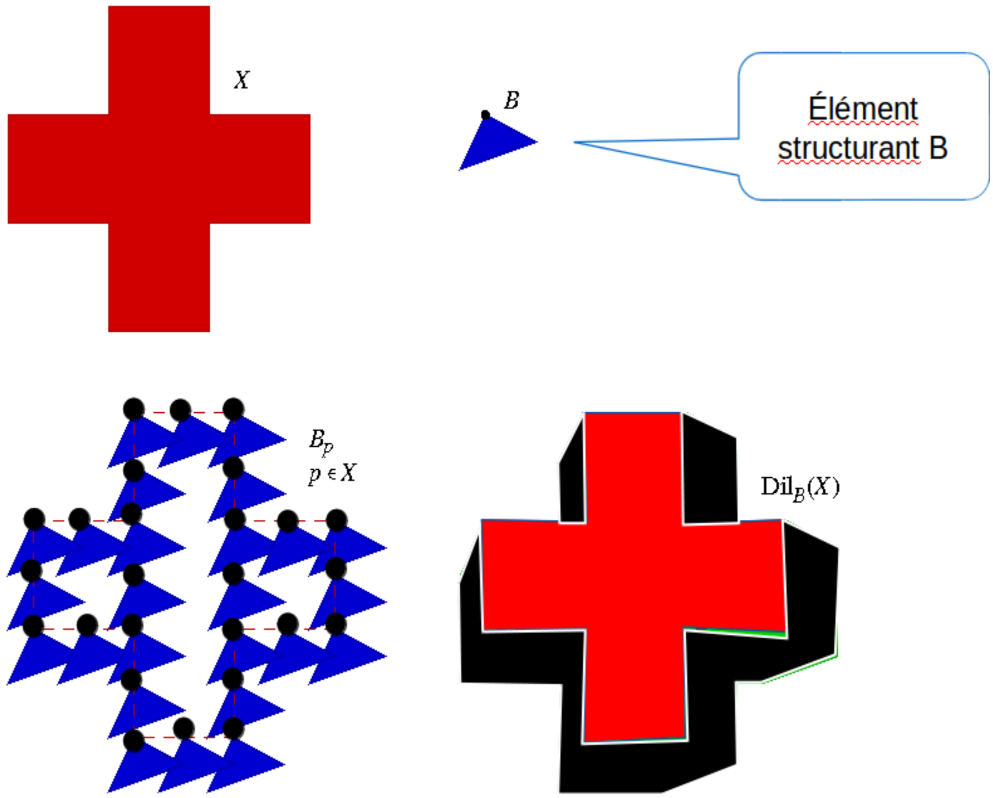


Figure 2 : un exemple de dilatation

Ainsi la dilatation consiste à étendre un objet comme le montre la figure ci-dessous :

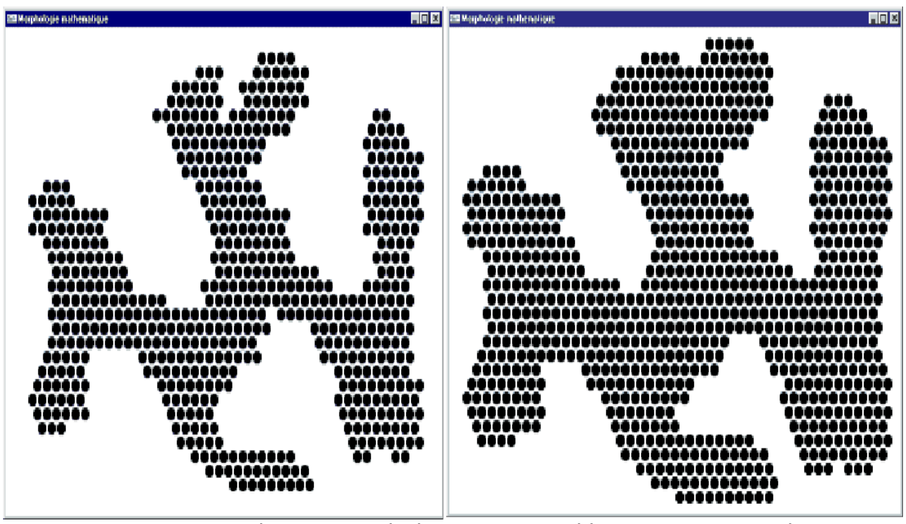


Figure 3 : agrandissement de l'image par dilatation (à gauche : l'image initiale, à droite : l'image finale)

L'Erosion

L'effet de l'érosion est d'abord de rétrécir la figure. La hauteur et largeur de la figure érodée seront les différences respectivement des hauteurs et largeurs de la figure originelle et de l'élément structurant. Si l'élément structurant est décentré, l'érosion décalera la figure en sens inverse. Enfin les coins concaves de la figure seront déformés en fonction de l'élément structurant (par exemple si celui-ci est un disque, les coins concaves sont arrondis).

Soit X une figure et B un élément structurant. L'érosion de X par B est l'ensemble des pixels p tels que la fenêtre Bp est incluse dans X :

$$\text{Eros}_B(X) = \{p \mid B_p \subseteq X\}.$$

La figure suivante montre une illustration:

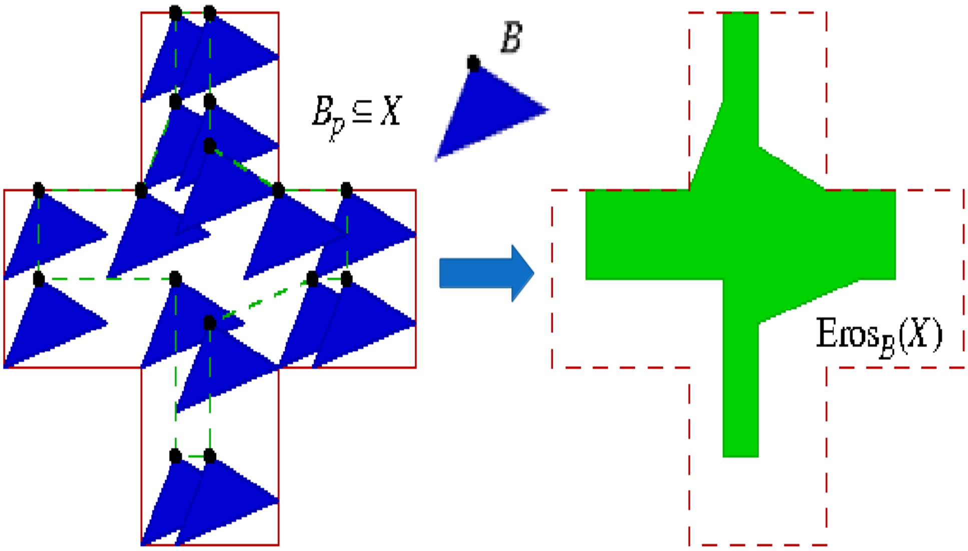


Figure 4 : un exemple de l'opération Erosion

L'érosion permet de réduire la taille ou le volume occupé par une région comme le montre la figure ci-dessous :

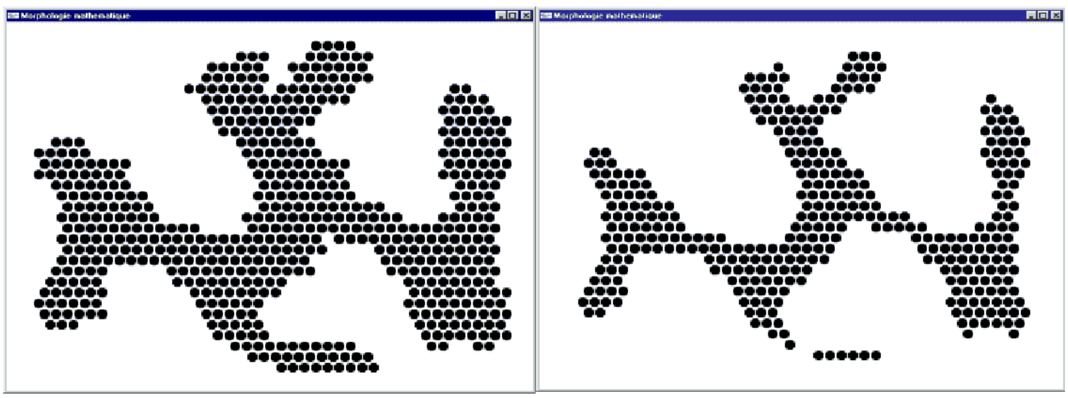


Figure 5 : réduction de la taille d'une région par l'érosion (à gauche :

l'image initiale, à droite : l'image finale)

L'ouverture

L'ouverture par B se définit comme la composition de l'érosion par B suivie de la dilatation par B : OuvB(X) = DilB(ErosB(X)). La figure suivante montre un exemple d'ouverture

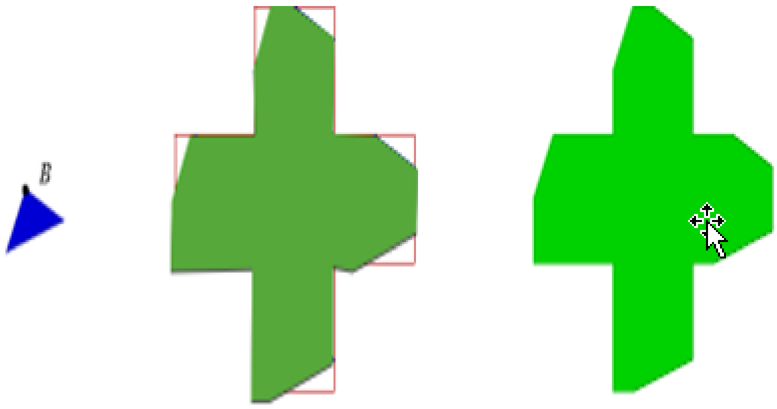


Figure 6 : une ouverture

Le but de l'ouverture est d'isoler les surfaces présentes dans l'image et de lisser les contours ; un autre exemple de l'ouverture est présenté par la figure ci-dessous.

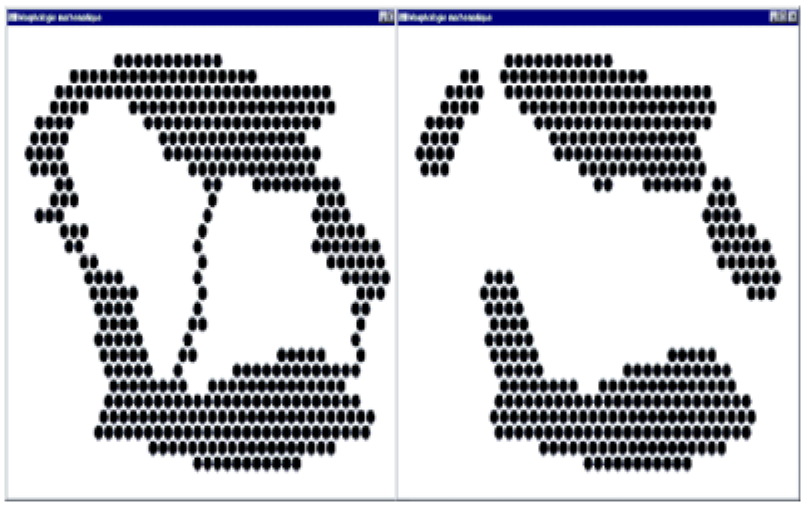


Figure 7: un autre exemple de l'ouverture (à gauche : l'image initiale, à droite : l'image finale)

La fermeture

La fermeture par B est définie comme la composition de la dilatation par B suivie de l'érosion par B : FermB(X) = ErosB(DilB(X)).

La fermeture vise recoller des morceaux de surfaces proches de manière à fermer des contours disjoints et à lisser les contours. La figure ci-dessous fournit un exemple de fermeture sur une image :

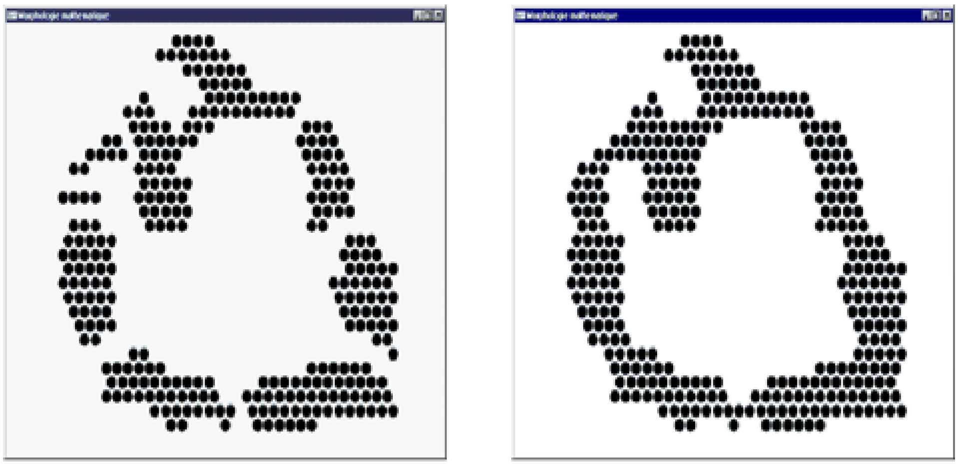


Figure 8 : un exemple de fermeture (à gauche : l'image initiale, à droite : l'image finale)

Notons que le logiciel GIMP donne la possibilité de dilater et érodé une image.

La squelettisation

La squelettisation d'un objet (une image) permet d'obtenir une version simplifiée d'un objet conservant certaines caractéristiques :

- le squelette à la même topologie que l'objet initial (il permet donc notamment d'isoler des composantes connexes);
- le squelette est moins lourd que l'objet initial (il permet donc la compression de données).

L'objectif de la squelettisation est de représenter un ensemble avec un minimum d'information (compression de données) sous une forme qui soit à la fois plus simple à extraire et commode à manipuler. Le squelette peut servir en reconnaissance de forme. Un exemple de squelettisation d'une image est proposé par la figure ci-dessous dans un espace euclidien :

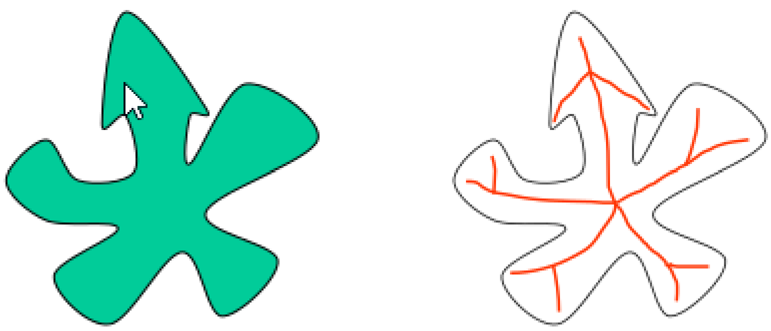


Figure 9 : squelettisation de l'objet à gauche donnant le squelette, un autre objet à droite

L'amincissement séquentiel permet d'obtenir le squelette d'un objet par composition des opérations d'érosion succéssive avec des éléments structurants différents. L'amincissement séquentiel de X par l'élément structurant B est défini comme :   AminSeqBi(X) = [(ErosB (ErosB (…))]

La figure suivante fournit un exemple d'amincissement séquentiel :

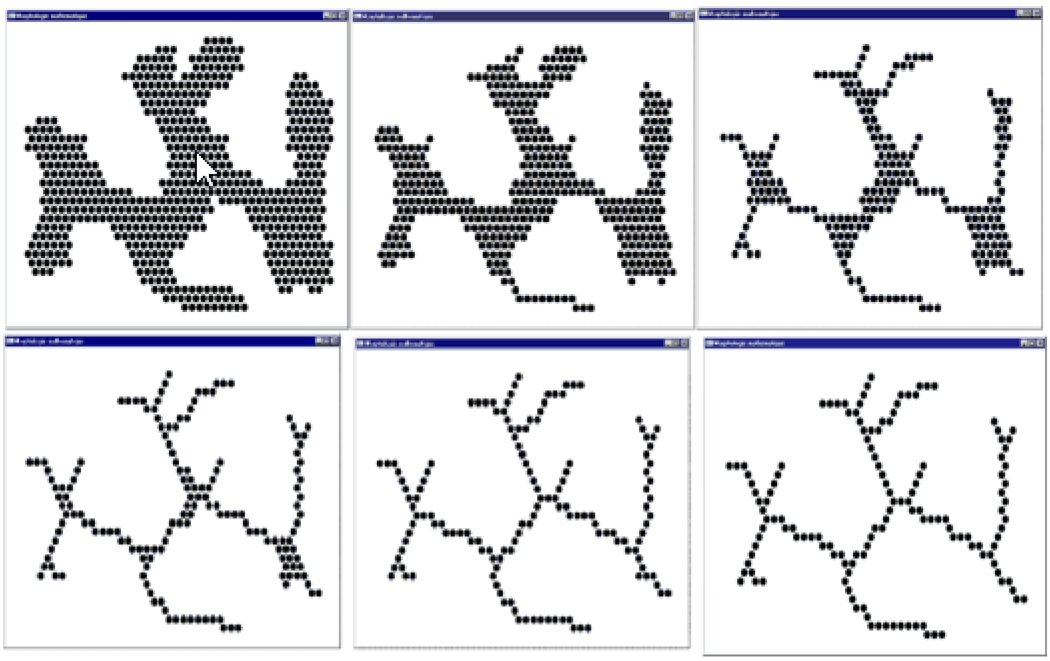


Figure10 : un exemple de squelettisation d'un objet (une image)

La squelettisation possède certaines propriétés dans un espace euclidien qui est rercherchées.

Quelques propriétés de la squelettisation dans un espace euclidien

Soient X un objet et S(X) le squelette de X.

- Réversibilté : le S(x) doit permettre de retrouver X: La figure ci-dessous montre une construction d'un objet à partir de son squelette.

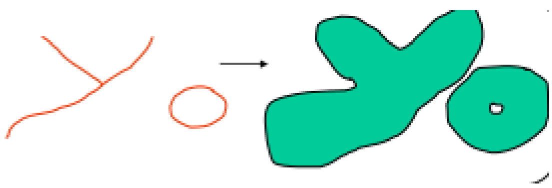


Figure 11 : un exemple de construction de X à partir de S(X)

- Préservation de la géométrie : le squelette s(X) doit permet d'avoir une connaissance de la forme géométrique de X. La figure ci-dessous fournit une illustration :

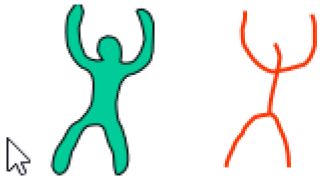


Figure 12 : forme géométrique

- Épaisseur nulle : le squelette S(X) doit être constitué de courbe sans épaisseur. Cela est présenté par la figure ci-dessous, où nous avons deux figures.

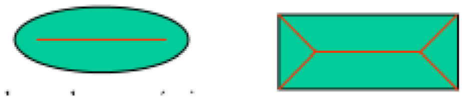


Figure 13 : épaisseur nulle

- Préservation de la topologie : lorsque un objet est composé de plusieurs objets connexes. Le squelette doit préserver cette connexité. La figure ci-dessous présente une image composée de plusieurs objets.

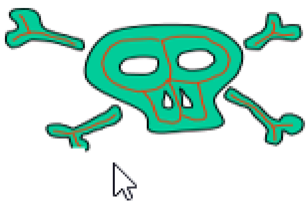


Figure 14 : un objet composé

- Invariance aux transformations affines : les rotations, les homothéties, les translation ne changent pas le squelette. La figure ci-dessous fournit un exemple.

Figure 15 : un objet soumis à des transformations

- Continuité : toute petite modification de la forme originale entraîne une modification du squelette. La figure ci-dessous fournit une illustration.

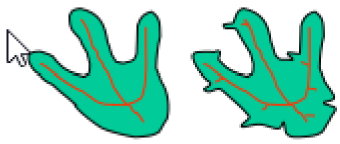


Figure 16 : un objet original modifié avec son squelette

Conclusion

Le principal domaine d'application de la morphologie mathématique est le traitement d'images. Elle fournit, en particulier, des outils de filtrage, de segmentation. Les images obtenues permettent également de construire des effets speciaux

Évaluation

Exercice : définir les termes suivants : ouverture, fermeture, dilatation, érosion et donner leur définition mathématique

Résumé de l'unité

En traitement d'images, les opérations de segmentation, de filtrage et de morphologie sont souvent utilisés soit pour améliorer la qualité de l'image, soit pour mettre en lumière certaines caractéristiques, soit pour réaliser des effets spéciaux.

Dans la segmentation, il ya trois approches (l'approche par contour, l'approche par régions, l'approche par seuillage).

Ils existent plusieurs filtres permettant soit de créer du bruit, soit de réduire le bruit, soit de mettre en lumière les contours d'une image.

Dans la morphologie, les définitions de l'érosion, la fermeture, l'ouverture, la dilation, la squelettisation permettent aussi de faire apparaître certaines caractéristiques d'une image.

Évaluation de l'unité

Vérifiez votre compréhension!

Exercice : On considère une image I(x, y) en dimension 2

1. Proposer un algorithme qui permet d'obtenir les contours Laplacien de cette image;
2. À partir de l'image obtenue, proposer un algorithme qui permet de déterminer l'histogramme des niveaux de gris.

Lectures et autres ressources

Les lectures et autres ressources de cette unité se trouvent au niveau des lectures et autres ressources du cours.

Unité 3: Transformations des Images

Introduction à l'unité

Dans cette unité, nous définissons, quelques transformations telles que la rotation (pivoter), l'homothétie, la translation, la projection et la transformée de Hough.

Nous entendons par transformation, l'application d'une fonction pouvant changer la position d'un pixel, changer la taille d'une image, ou la forme d'une image.

Objectifs de l'unité

À la fin de cette unité, vous devriez être capable de:

1. Pivoter une image ;
2. Agrandir ou de réduire la taille d'une image ;
3. Déplacer une image à une position donnée ;
4. Maîtriser ces opérations en dimension 2 et 3 ;
5. Comprendre la transformée de Hough.

Termes clés

Translation : une translation d'une image consiste à déplacer une image suivant une vecteur directeur

Rotation : Une rotation en infographie se traduit souvent par un pivotement. La rotation consiste alors à tourner une image suivant un angle de rotation et par rapport à un centre (centre de gravité de l'image par exemple)

Homothétie : L'homothétie est l'opération qui consiste à agrandir ou à réduire la taille d'une image.

Activités d'apprentissage

Activité 1 - Translations, rotations et homothétie

Introduction

Dans cette activité, on présentera trois transformations, souvent utilisées en infographie, pour traiter les images : la rotation , la translation, et l'homothétie. Ces transformations sont définies mathématiquement en faisant intervenir des matrices.

Détails de l'activité

Les données d'une image sont représentées dans un repère comme le montre la figure ci- dessous : chaque pixel est associé à des coordonnées dans un repère OXY. Cela permet d'appliquer aisément la définition mathématique des transformations

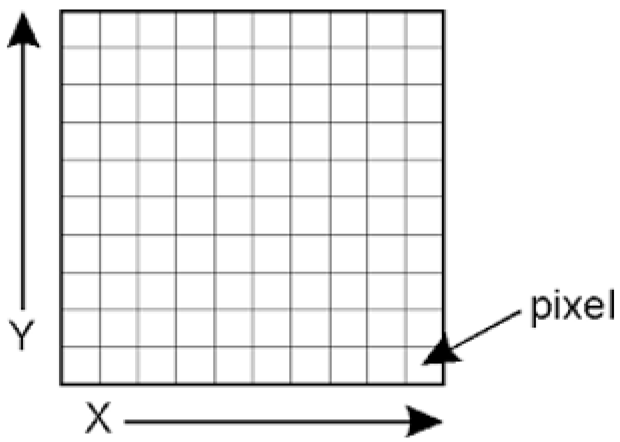


Figure 1 : des pixels dans une image

[Source: http://nmita.iowa.uiowa.edu/instr_co.htm]

Rotation: La rotation consiste à tourner l'image suivant un sens de rotation défini par un angle et par rapport un centre de rotation. En infographie, la rotation d'un rectangle ou d'un carré suivant un angle donné, par rapport à leur centre de gravité est souvent utilisée. Le terme pivoter est employé dans ce cas.

Soit A (x1 y1) un point dans un repère classique OXY orthonormé. La rotation du point A suivant un angle θ, par rapport à l'origine du repère donne un point B (x2 y2) dont les coordonnées sont définies par :

$$x2 = x1\cos\theta + y1\sin\theta$$

$$y2 = x1\sin\theta - y1\cos\theta$$

La matrice de la rotation est donc définie par :

[x2 y2 1] = [x1 y1 1] *

Comme nous travaillons dans un espace discret où les coordonnées sont des points de Z x Z c'est à dire des valeurs entières, le problème avec la rotation, les points B (x2 y2) ne sont pas toujours des points de Z x Z. Il apparaît donc intéressant de proposer une rotation discrète.

Translation : la translation consiste à déplacer une image dans un espace suivant un vecteur directeur. Soit un point P(x1, y1) à translater pour obtenir un point Q(x2, y2). Si le vecteur directeur de la translation a pour coordonnées (Tx , Ty), les coordonnées de Q sont alors déterminées par la relation x2 =x1 +Tx et y2 = y1+ Ty.

Par exemple: Si A(20,10), B(30,100) et C(40,70) sont les sommets d'un triangle translaté par le vecteur directeur de coordonnées (20, 10), le nouveau triangle obtenu possédera comme les trois sommets A1(20+20, 10+10) B1 (30+20, 10+10) C1(40+20, 70+10). La forme de la matrice de la translation est donc :

[x2 y2 1] = [x1 y1 1] *

Homothétie à coefficients variables (changement d'échelles) : l'homothétie permet de réduire ou d'agrandir la taille d'une image donnée. Cette transformation est déterminée par des coefficients qui indiquent s'il y a une réduction ou un agrandissement. Cette opération est très souvent réalisée en infographie ou nous pouvons agrandir ou réduire une image suivant un axe donné. Soit un point A (x1 y1) qui doit être transformée pour donner un point B (x2, y2). Suivant les facteurs (k1, k2) respectivement pour l'axe des abscisses x et l'axe des ordonnées y . Nous avons les coordonnées de B déterminées par x2 = x1 * k1 et y2 = y1 * k2.

La matrice M associée au changement d'échelles est alors :

$$M = \begin{pmatrix} k1 & 0 \\ 0 & k2 \end{pmatrix}$$

L'homothétie de facteur constant k est définie par la matrice M :

$$M = \begin{pmatrix} k & 0 \\ 0 & k \end{pmatrix}$$

La symétrie d'axe OX est définie par la matrice M :

$$M = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

La figure montre un exemple de rotation, de translation et d'homothétie.

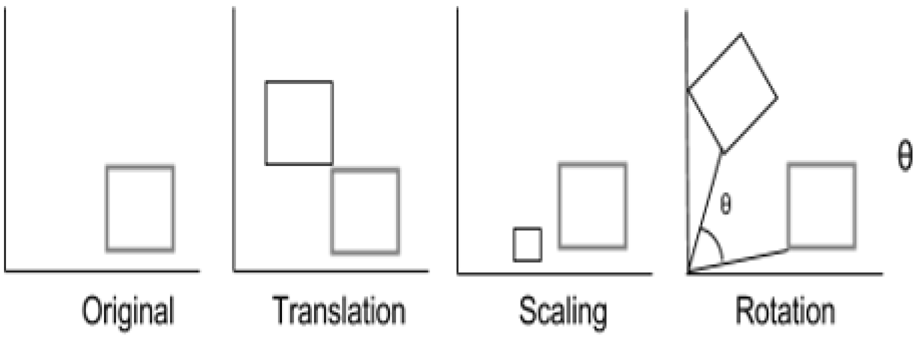


Figure 2 : les types de transformations

[Redrawn from: http://www.pling.org.uk/cs/cgv.html]

En dimension 3, il est possible d'étendre ces transformations. Pour la rotation, elle se fera sur les axes OX, OY, OZ du repère classique OXYZ. Les matrices M correspondantes à ces transformations 3D sont :

$$\textit{translation de vecteur } \boldsymbol{U}(M,N,P) : \mathrm{M(U)} = \begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ M & N & P & 1 \end{matrix}$$

$$\textit{rotation d'angle } \alpha \textit{ autour de } Ox : \mathrm{M}(x_\alpha) = \begin{matrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

$$\textit{rotation d'angle } \beta \textit{ autour de } Oy : \mathrm{M}(y_\beta) = \begin{matrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

$$\textit{rotation d'angle } \gamma \textit{ autour de } Oz : \mathrm{M}(z_\gamma) = \begin{matrix} \cos\gamma & \sin\gamma & 0 & 0 \\ -\sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

$$\textit{dilatation de coefficients } E,\ F,\ G : \mathrm{M(E,F,G)} = \begin{matrix} E & 0 & 0 & 0 \\ 0 & F & 0 & 0 \\ 0 & 0 & G & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

Ces transformations peuvent être combinées : on obtient ainsi une combinaison de matrices. Par exemple, on pourrait avoir une translation suivie d'une rotation ou réciproquement.

De plus, il existe également d'autres transformations ;

- La transformation du photomaton : le principe de la transformation du photomaton suit le schéma suivant :

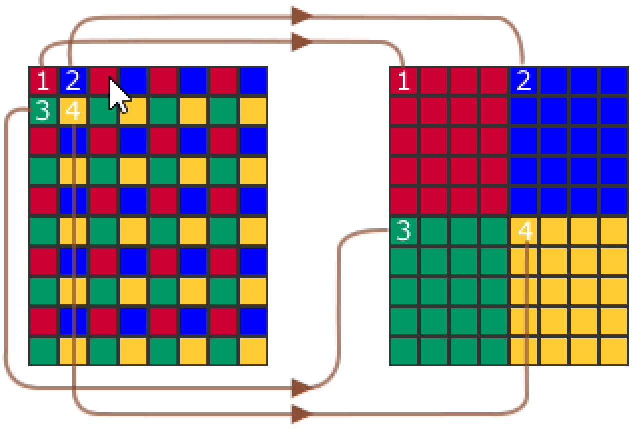


Figure 3 : la méthode de la transformation du photomaton

Voici quelques illustrations de la transformation du photomaton dans la figure ci-dessous :

Figure 4 : la méthode de la transformation du photomaton

- Des transformations non affines:

Formule :

$$\varphi : \begin{cases} x' = \frac{2r}{\pi} \sin\left(\frac{\pi x}{2r}\right) \\ y' = y \end{cases}$$

Exemple de résultat :

Figure 5 : la méthode de la transformation du photomaton

Formule :

$$\varphi : \begin{cases} x = \dfrac{2r}{\pi}\arcsin\left(\dfrac{x'}{r}\right) \\ y = y' \end{cases}$$

Exemple de résultat :

Figure 6 : la méthode de la transformation du photomaton

- La transformation de Hilbert permet également de réaliser des effets spéciaux. Dans la transformation de Hilbert, l'image est transformée de telle sorte que le pixel 1 va occuper la place du pixel 2, le pixel 2 va occuper la place du pixel 3, etc. Le dernier pixel revient prendre la place du pixel 1 comme présentée dans la figure ci-dessous.

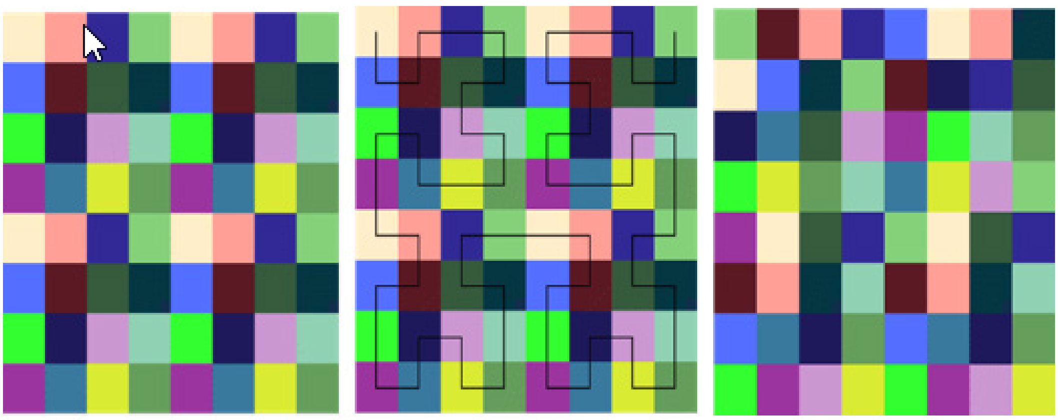


Figure 7 : la méthode de la transformation de Hilbert

Conclusion

Cette activité a permi de définir et d'illustrer trois transformations basiques très utilisées en infographie telles que la rotation, la translation et l'homothétie en dimension 2 et 3.

Évaluation

Exercice 1 : proposer un algorithme qui permet de réaliser la rotation d'une image quelconque 2D.

Exercice 2 : proposer un algorithme qui permet de réaliser la translation d'une image quelconque 2D

Exercice 3 : proposer un algorithme qui permet de réaliser le changement d'échelles d'une image quelconque 2D

Exercice 4 : proposer un algorithme qui permet de réaliser la transformation du photomaton.

Exercice 5 : proposer un algorithme qui permet de réaliser la transformation de Hilbert.

Activité 2 - Projections

Présentation

Dans cette activité, nous étudierons les différents type de projections en fournissant des illustrations.

Détails de l'activité

Il existe essentiellement deux classes de projection : les projections perspectives et les projections en paralèlle. Les projections perspectives où les lignes de projection convergent en un point appelé centre de projection et les projections parallèles où le centre de projection est rejeté à l'infini et où les lignes de projection sont des droites parallèles.

Ces deux types de projections sont illustrés par la figure ci-dessous, la projection perspective en (a) et la projection paralèlle en (b) :

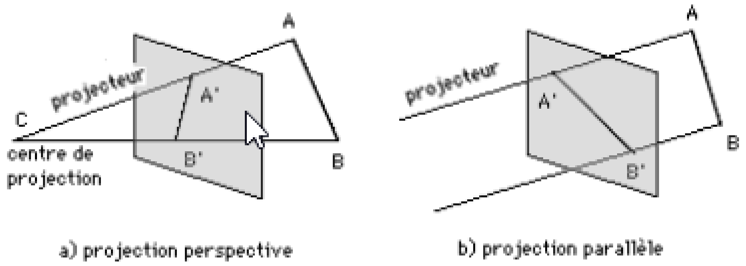


Figure 1 : un exemple de projection perspective en (a) et un exemple de projection paralèlle en (b)

On peut avoir plusieurs points de centre de projection dans le cas de la projection perspective comme le montre la figure suivante :

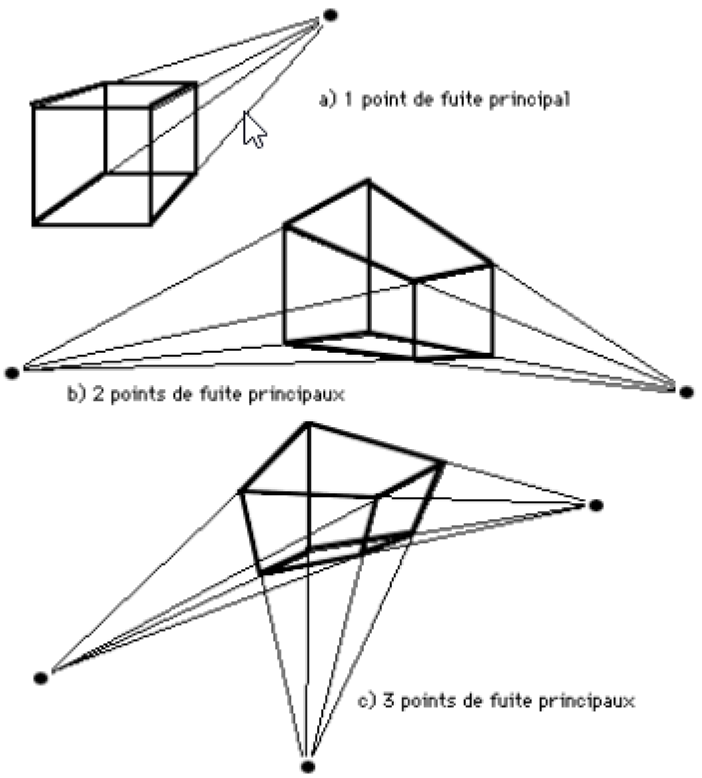
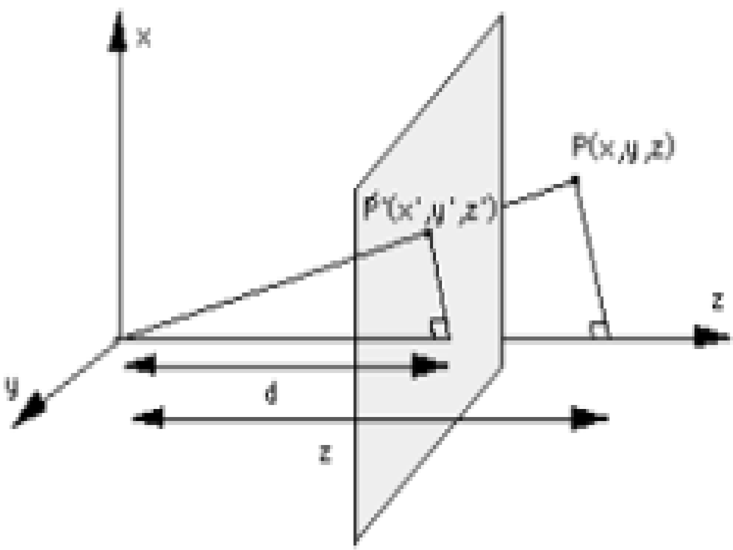


Figure 2 : un exemple de projection perspective et un exemple de projection paralèlle

Matrice de la projection perspective

Dans la projection perspective, la relation en un point P(x, y, z) et P'(x', y', z') son projété en perspective suit les propriètés du théorème de Thalès : la figure ci-dessous fournit une illustration de cette relation



--

Figure 3 : un exemple de projection perspective et un exemple de projection paralèlle

Nous avons les relations, d'après Thalès : x'/d = x/z et y'/d = y/z d'où x'=x/(z/d), y' = y/(z/d) et z' = d. Et finalement [x' y' z' 1] = [x/(z/d) y/(z/d) d 1]

D'où la relation :

$$\begin{matrix} x & y & z & z/d \end{matrix} = \begin{matrix} x & y & z & 1 \end{matrix} \cdot \begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1/d \\ 0 & 0 & 0 & 0 \end{matrix}$$

Conclusion

Nous avons présenté brièvement dans cette activité les différents types de projection. Cela permet d'avoir des notions de base théoriques sur la projection et de pouvoir par exemple utiliser le logiciel POV-RAY en perspective pour mettre en œuvre ces projections.

Évaluation

Exercice 1 : définir les termes suivants et donner des exemples

1. Projection perspective ;
2. Projection paralèlle.

Exercice 2 : proposer la matrice dans le cas de la projection paralèlle d'un point.

Activité 3 -Transformées de Hough

Introduction

La transformée de Hough a été proposé en 1962 par Paul Hough. Cette méthode est utilisée en analyse d'images pour déterminer des segments de droites discrètes dans une image.

La transformée de Hough utilise deux espaces : l'espace image qui constitué par l'image initiale représenté dans un repère; l'espace de paramètres également appélé accumulateur est un tableau ou une matrice. Plusieurs variantes de la transformée de Hough ont été proposées. Nous présenterons la transformée de Hough pour la détéction des droités discrètes et la transformée de Hough pour la détection des cercles discrets. Dans la librairie Opencv, ils existent des fonctions permettant de réaliser la reconnaissance dans une image d'une droite discrète et d'un cercle discret.

Détails de l'activité

La transformée de Hough pour la détection des droites discrètes

La transformée de Hough qui associe un point continu de l'espace image à une droite continue de l'espace de paramètres est définie par :

Définition *(Transformée de Hough d'un point continu) Soient I_2 un espace image ($\subset \mathbb{R}^2$) et P un espace de paramètres ($\subset \mathbb{R}^2$). Soient f et g deux fonctions définies par :*

La fonction f associe un point de I_2 à une droite de P_2

$$f : \begin{array}{ccc} I_2 & \rightarrow & P_2 \\ M(x_0, x_1) & \mapsto & \{(y_0, y_1) \in P_2 / y_1 = x_1 - x_0 y_0\} \end{array}$$

La fonction g associe un point de P_2 à une droite de I_2

$$g : \begin{array}{ccc} P_2 & \rightarrow & I_2 \\ N(y_0, y_1) & \mapsto & \{(x_0, x_1) \in I_2 / x_1 = y_1 + x_0 y_0\} \end{array}$$

Les images f(M) et g(N) sont appelées respectivement les transformées de Hough de M et N.

Ainsi un point dans l'espace de paramètres représente les coordonnées ou les paramètres d'une droite dans l'espace image. En application dans un espace discret, les points continus deviennent des points discrets avec des approximations (arrondis).

La figure donne un exemple de cette relation duale entre un espace image et un espace de paramètres.

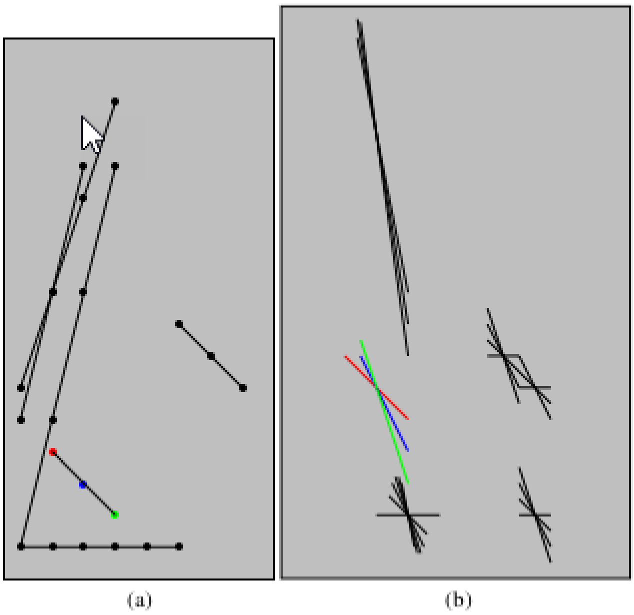


Figure 1 : relation duale entre un espace image en (a) et un espace de paramètres en (b)

Une application de la transformée de Hough sur une image réelle est fournie par la figure ci-dessous.

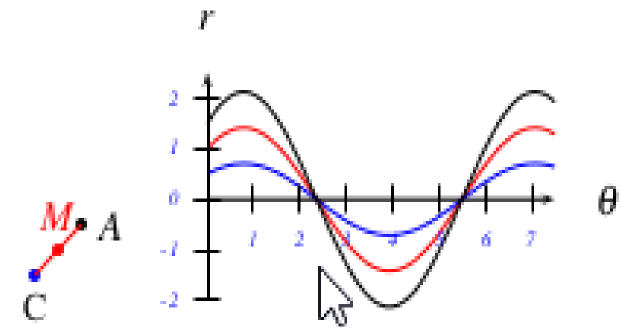
(a) (b)

Figure 2 : résultat de la transformée de Hough appliquée à une image en utilisant la librairie Opencv

Une autre variante de la transformée de Hough appéllée transformée de Hough standard associe un point de l'espace image à une courbe sinusoïdale. Cette transformée est définie par :

Définition *Soient I et P respectivement un espace image sous ensemble de $\mathbb{R}^2$ et un espace de paramètres sous ensemble de $\mathbb{R}^2$. Soit M(x, y) un point de I. La transformée de Hough standard de M est l'ensemble des points de $S(M)$ dans P défini par* $S(M) = \{(\theta, r) \in \rho^2 / r = x * cos\theta + y * sin\theta\}$

La figure suivante donne un exemple de transformée de Hough standard d'un ensemble de trois points A, M, .



(a) *[AC]* (b) *dual de A, M, C*

Figure 3 : Transformée de Hough Standard de trois points A, M, C

La transformée de Hough pour la détection des cercles discrets

La transformée de Hough pour la détection des cercles discrets utilise la définition de la transformée de Hough paramétrée qui est basée sur l'équation mathématique des formes géométriques : certaines variables sont considérées comme des variables de l'espace image et d'autres, des variables de l'espace de paramètres. la reconnaissance du cercle d'équation prend en compte la reconnaissance d'arcs de cercle avec a, b, r des variables de l'espace de paramètres et (x, y) des variables de l'espace image. A chaque point (x i , y i) du cercle continu dans l'espace image est associé, le cercle defini par dans l'espace de paramètres. La librairie Opencv propose une fonction permettant la reconnaissance d'arce de cercles discrets

D'autres variantes de la transformée de Hough pour la detection des ellipses ou des formes quelconques (appéllées la transformée de Hough généralisée, proposée par Ballard).

Conclusion

Nous avons présenté dans cette activité la méthode de la transformée de Hough permettant d'extraire des segments de droites et des arcs de cercles dans une image.

Évaluation

Exercice 1: définir les termes suivants :

1. Espace image;
2. Espace de paramètres.

Exercice 2 : proposer un algorithme de reconnaissance de droites discrètes à l'aide de la transformée de Hough

Résumé de l'unité

Les principales transformations des images sont la rotation, le changement d'échelle, la translation, l'homothétie. D'autres transformations comme la transformation de Hilbert et la transformation du photomaton permettent de représenter l'image sous une autre forme en changéant son contenu.

Évaluation de l'unité

Vérifiez votre compréhension!

Lectures et autres ressources

Les lectures et autres ressources de cette unité se trouvent au niveau des lectures et autres ressources du cours.

Unité 4: Etudes de Cas Sur Des Images 2D et 3D

Introduction à l'unité

Dans cette unité, nous allons mettre en œuvre les concepts définis dans les unités précédentes. Il s'agit de regarder dans la librairie OPENCV et de répondre aux questions :

1. Comment peut-on implémenter précisément la notion d'une image et les Concepts de droites discrètes, les courbes discrètes en dimension 2 et 3?
2. Comment peut-on implémenter les transformations des images (rotation, translation, homothétie, transformée de Hough) ?
3. Comment peut- on implémenter le filtrage, la segmentation et la morphologie?

D'autre part pour ces questions, nous montrerons à travers le logiciel GIMP, comment ces questions sont déjà traitées.

Objectifs de l'unité

À la fin de cette unité, vous devriez être capable de:

1. Comprendre la représentation d'une image à travers la librairie OPENCV ;
2. Traduire les notions de courbe discrète en dimension 2 et 3 à l'aide de la librairie OPENCV;.
3. Traduire les notions sur les transformées des images utilisant les types de données et les fonctions disponibles dans la librairie OPENCV ;
4. Traduire les notions sur la segmentation, le filtrage, la morphologie en utilisant les types de données et les fonctions disponibles dans la librairie OPENCV ;
5. identifier et utiliser quelques concepts du traitement d'image dans un logiciel de traitement d'images comme GIMP

Activités d'apprentissage

Activité 1 - Notion d'images et courbes discrètes

Introduction

Dans cette activité, nous présentons des exercices pratiques permettant de maîtriser la notion d'images et courbes discrètes en utilisant la librairie OpenCV.

Détails de l'activité

1. Objectives : savoir traduire les définitions de droites discrètes en un programme tracé de droites en C++
2. Ressources requises : CodeBlock, OPENCV, CIMG.H, POVRAY
3. Temps nécessaire : l'installation de OPENCV peut prendre du temps, cela dépend du débit internet : soient 24 heures pour installer et comprendre quelques fonctions de OPENCV et 3h pour installer et comprendre quelques fonctions de CIMG.H et 10 heures pour faire les travaux pratiques
4. Description des travaux pratiques :

   a. Donner deux points discrets en OPENCV.

   b. Proposer une fonction en C++ qui trace une droite discrète de Brésenham reliant ces deux points.

   c. Proposer une fonction en C++ qui trace une droite discrète standard de Reveillès reliant ces deux points.

   d. Proposer un programme en C++ qui trace un cercle discret de Reveillès en utilisant la librairie OPENCV.

   e. La librairie OPENCV propose des fonctions de tracé de droites, de rectangles, de cercles, donner des exemples d'utilisation de ces fonctions.

   f. Répondre aux questions précédentes en utilisant la librairie CIMG.H

   g. Générer avec POVRAY des objets en dimension 2, et en dimension 3

- Préréquis : cours théoriques sur les images et les droites discrètes et les courbes discrètes.
- References : voir les liens http://www.codeblocks.org/ pour télécharger codeblock, http://opencv.org/ pour télécharger OPENCV, http://cimg.eu/download.shtml pour télécharger CIMG.H, http://www.povray.org/ pour télécharger POVRAY

Conclusion

Dans cette activité, le travail a consisté à tracer des droites discrètes avec les librairies OPENCV et CIMG.H.

Activité 2 - Les transformations des images

Présentation

Dans cette activité, il s'agit de montrer à l'étudiant concrètement comment il peut faire des rotations, des translations, des agrandissements ou des réductions de la taille sur une image donnée.

Détails de l'activité

1. Objectives : savoir appliquer des transformations par un programme en C++ et l'utilisation d'un logiciel comme GIMP
2. Ressources requises : CodeBlock, OPENCV, CIMG.H, POVRAY, GIMP
3. Temps nécéssaire : l'installation de OPENCV peut prendre du temps, cela dépend du débit internet : soient 24 heures pour installer et comprendre quelques fonctions de OPENCV et 3h pour installer et comprendre quelques fonctions de CIMG.H et 10 heures pour faire les travaux pratiques.
4. Description des travaux pratiques :

    a. Donner un exemple de rotation sur une image, en OPENCV.

    b. Donner un exemple de changement d'échelles (homothétie à coefficients variables) sur une image, en OPENCV.

    c. Répondre à ces deux questions précédentes en utilisant CIMG.H

    d. Répondre à ces deux questions (1 et 2) sans utiliser une librairie.

    e. Avec le logiciel GIMP, il est possible d'effectuer une rotation, des changements d'échelles, notamment avec GIMP 2.8.10, en allant dans Tool->Transform Tools, vous trouverez les transformations. Proposer des exemples de rotations et de changements d'échelles (scale).

    f. La librairie OPENCV propose des fonctions pour la transformée de Hough, proposer un exemple de code pour réaliser la détection de segments de droites dans une image. Proposer également un exemple de code pour réaliser la détection de cercles ou d'arc de cercles droites dans une image.

g. Donner des exemples de transformations en utilisant le logiciel POVRAY.

h. Faire des projections d'objets 3D en 2D en utilisant le logiciel POVRAY.

- Préréquis : cours théoriques sur les transformations des images
- References : voir les liens http://www.codeblocks.org/ pour télécharger codeblock, http://opencv.org/ pour télécharger OPENCV, http://cimg.eu/download.shtml pour télécharger CIMG.H, http://www.povray.org/ pour télécharger POVRAY

Conclusion

Dans cette activité, le travail a consisté à appliquer concrètement des transformations des images, avec les fonctions disponibles dans les librairies OPENCV et CIMG.H et avec le logiciel GIMP.

Activité 3 -La segmentation, le filtrage et la morphologie

Introduction

Dans cette activité, il s'agit de montrer concrètement à l'étudiant comment peut on segmenter, filtrer ou appliquer les opérateurs de morphologie vue pendant le cours théorique.

Détails de l'activité

1. Objectives : savoir filtrer et appliquer les opérateurs de morphologie sur une image en utilisant un logiciel de traitement d'image comme GIMP ; savoir écrire un programme en C++ qui implémente un filtre.
2. Ressources requises : CodeBlock, OPENCV, CIMG.H, GIMP
3. Temps nécéssaire : l'installation de OPENCV peut prendre du temps, cela dépend du débit internet : soient 24 heures (en perso) pour installer et comprendre quelques fonctions de OPENCV et 3h (en perso) pour installer et comprendre quelques fonctions de CIMG.H et 10 heures pour faire les travaux pratiques.
4. Description des travaux pratiques :

a. 1. Proposer un exemple de programme en C++ qui génère un histogramme des couleurs en utilisant OPENCV.

b. 2. Implémenter le filtre Sobel, le filtre moyenne et le Laplacien en C++ avec OPENCV.

c. 3. Répondre à ces deux questions précédentes en utilisant CIMG.H

d. 4. Avec le logiciel GIMP, il est possible de créer son propre filtre, notamment avec GIMP 2.8.10, en allant dans Filters->Generic->Convolution Matrix. Proposer deux exemples de filtrage d'image.

e. Donner des exemples de transformations en utilisant le logiciel POVRAY

- Préréquis : cours théoriques sur la segmentation, le filtrage, la morphologie
- References : voir les liens http://www.codeblocks.org/ pour télécharger codeblock, http://opencv.org/ pour télécharger OPENCV, http://cimg.eu/download.shtml pour télécharger CIMG.H,

Conclusion

Dans cette activité, le travail a consisté à manipuler des transformations des images avec les fonctions disponibles dans les librairies OPENCV et CIMG.H et avec le logiciel GIMP.

Résumé de l'unité

Les cas concrets de toutes les unités des cours théoriques ont été présentés dans cette unité, allant à des réalisations de programme en c++ utilisant les librairies OPENCV,CIMG.H, à la manipulation des logiciels GIMP, POVRAY pour tracer des courbes discrètes, faire des transformations sur des images et filtrer des images.

Évaluation de l'unité

Vérifiez votre compréhension!

Lectures et autres ressources

Les lectures et autres ressorces de cette unité se trouvent au niveau des lectures et autres ressources du cours.

Siège de l'Université Virtuelle Africaine

The African Virtual University
Headquarters

Cape Office Park

Ring Road Kilimani

PO Box 25405-00603

Nairobi, Kenya

Tel: +254 20 25283333

contact@avu.org

oer@avu.org

Bureau Régional de l'Université Virtuelle Africaine à Dakar

Université Virtuelle Africaine

Bureau Régional de l'Afrique de l'Ouest

Sicap Liberté VI Extension

Villa No.8 VDN

B.P. 50609 Dakar, Sénégal

Tel: +221 338670324

bureauregional@avu.org