

SIMES - 961620
Système d'Information Multimédia
Pour l'Environnement Subsaharien

Parameters extraction

Deliverable number : D12

Nature:P

Contractual Date of Delivery: may 9, 2001

Task WP2.2 : Parameters extraction

Authors :

Hippolyte TAPAMO

Institute : Université de Yaoundé I

E-mail : Tapamo_Hippolyte@yahoo.com

Olivier MONGA

Institute : INRIA

E-mail : Olivier.Monga@inria.fr

ABSTRACT

The present document follows Deliverable D11 and presents how to use image processing algorithms results (regions, contours, typical points) in order to get semantical information relevant for end-users. This information could be for instance used as input for the Niger Central Delta integrated model presented in Deliverable D11.

This document constitutes deliverable D12.

Key Words : region, edge, thin network, road, river, water plain, rice-plantation, lineic.

1. Introduction

Ce document fait suite au livrable D11 et présente comment utiliser les résultats des algorithmes de traitement d'images (régions, contours, points caractéristiques) pour obtenir des informations sémantiques pertinentes pour les utilisateurs. Ces informations pourront par exemple être prises en compte comme entrées du modèle de gestion intégrée du Delta Central du Niger présenté dans le livrable D11.

Les images dans le projet SIMES sont d'une importance capitale. Elles représentent un vecteur essentiel d'informations. L'analyse de ces images permet aux acteurs en charge de la gestion de l'espace et de l'aménagement du territoire de localiser les zones d'intérêt et de suivre leur évolution au cours du temps.

Les images de la Terre, en prise de vue verticale (image satellitale ou issue d'une campagne aérienne classique) ou oblique (image prise du sol ou d'un aéronef léger), se caractérisent par la diversité de leurs modes d'enregistrement (verticale, oblique, altitudes diverses, multispectrale, panchromatique, radar, etc.) et par la masse considérable de données qu'elles représentent. Le problème se pose alors de l'automatisation de certaines tâches de repérage et de préparation à l'interprétation intelligente. Si l'opérateur humain reste le plus efficace pour extraire les informations qualitatives et sémantiques, il est parfois mal adapté pour l'extraction rapide d'informations quantitatives (mesures de superficies, dénombrement d'objets) et pour le traitement géométrique systématique des données (mosaïcage, recalage).

C'est pourquoi nous avons étudié comment extraire, à l'aide des techniques modernes de vision par ordinateur, des informations quantitatives et qualitatives des images en provenance du site pilote –le Delta Central du Niger-. Ces traitements ont aussi été essayés sur bien d'autres images pour accroître la base de tests. Comme exemple de traitement, un opérateur (utilisateur final) voudra par exemple savoir combien de rizières se trouvent dans une zone donnée ou encore quelle est la superficie d'une rizière qu'on trouve dans une zone sans avoir à se rendre forcément sur le terrain - ou encore localiser les zones inondées (bras du fleuve de largeur fluctuante, mares permanentes d'étendue fluctuante, ou zones temporairement inondées) et en tirer des informations quantitatives, comme la largeur d'un bras ou l'étendue d'une zone inondée. On voudra aussi par exemple estimer la longueur d'un bras ou d'une route dans une portion donnée de l'image, ou localiser et estimer l'étendue des rizières.

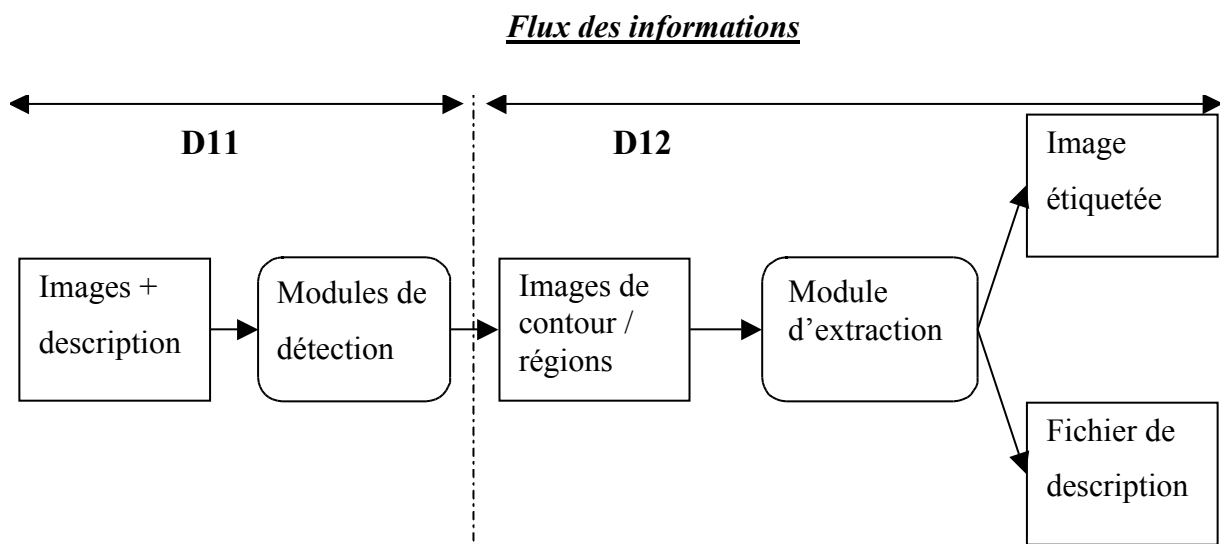
De manière générale, il s'agira d'extraire de ces images, les plantations telles les rizières, les structures linéiques telles que les routes et les cours d'eau, les structures surfaciques géométriquement irrégulières telles que les zones inondées. Les méthodes utilisées pour réaliser ce travail seront différentes selon que l'on veut extraire les structures linéiques, les rizières ou les zones inondées. Ceci nous conduira à la définition et à l'implémentation des algorithmes permettant d'extraire ces différents types de paramètres et de fournir les informations quantitatives et qualitatives afférentes. L'extraction des zones de riziculture sera basée sur l'approximation polygonale des contours de l'image par des rectangles tandis que celle des routes et des zones inondées sera faite en utilisant la géométrie différentielle.

2. Extraction des rizières

2.1. Description générale

Notre système prend en entrée une image (satellite, photo aérienne, ...) et un fichier descriptif de cette image. Les images sont généralement issues de capteurs divers et par conséquent sont de formats différents, de résolutions différentes, ... Dans le processus d'extraction des caractéristiques, la première étape est celle qui nous permet d'extraire les contours et/ou les régions de l'image en entrée.

A la suite de cette opération de segmentation des contours et/ou des régions, nous obtenons une ou plusieurs images et/ou un fichier d'informations supplémentaires. A partir de ces résultats, on applique les algorithmes qui vont être décrits dans la suite de ce rapport pour extraire les informations souhaitées.



Le fichier de description de la donnée image en entrée permet notamment au bout du processus d'extraction de fournir des informations quantitatives.

L'extraction des contours permet de faire ressortir les contours d'une image tandis que la méthode des régions nous permet d'en extraire des zones homogènes en terme de niveau de gris. Pour l'algorithme, une zone homogène est celle où l'on a une faible variation de niveau de gris. La notion de faible variation de niveau de gris étant relative, il apparaît donc que le résultat va dépendre du facteur qui va permettre de dire si oui ou non le niveau de gris a changé. Ce facteur est la valeur du gradient. Lorsqu'il est maximal, on considère qu'on se trouve sur un contour. Ce calcul du gradient dépend des valeurs des paramètres qu'on va passer à l'algorithme responsable de la segmentation.

Selon l'image et selon le résultat attendu d'une segmentation en régions ou en contours, il faut trouver les paramètres appropriés. C'est généralement de manière expérimentale que l'on détermine les meilleures valeurs des paramètres pour une opération donnée. C'est à dire qu'on réalise au départ plusieurs essais en modifiant à chaque fois les paramètres jusqu'à

obtenir un jeu de paramètres qui répond le mieux l'attente. Après cette étape de segmentation, on passe à l'étape de l'extraction plus reconnaissance des formes qui nous intéresse (les structures rectangulaires pour les rizières, les structures linéiques pour les routes et les cours d'eau, ...). La troisième étape va consister à évaluer quantitativement et qualitativement les formes extraites.

2.2. Méthode "contours"

Etapas du processus

- ***Détection des points de contours par calcul du gradient par filtrage puis extraction des maxima locaux du gradient***

Cette partie est réalisée avec la bibliothèque de routines de traitement d'images MegaWave2. Cette bibliothèque d'algorithmes de traitement d'image dispose d'une rubrique qui traite de la segmentation en contours, en régions et en texture des images. Dans cette rubrique on trouve le module `Segct` qui produit à la fois les contours, les régions et une structure de données appelée `Curves`. Cette structure de données est une liste de chaînes. Chaque chaîne représente une liste de coordonnées des points de contours de l'image segmentée. Nous obtenons ainsi une structure de données facilement manipulable.

- ***Chaînage puis approximation polygonale des points de contours***

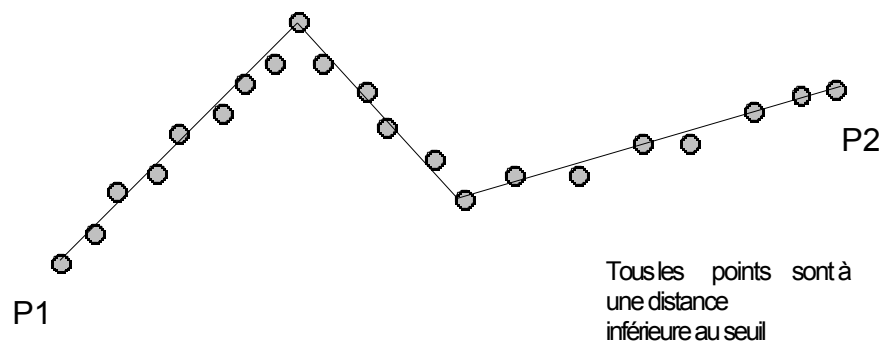
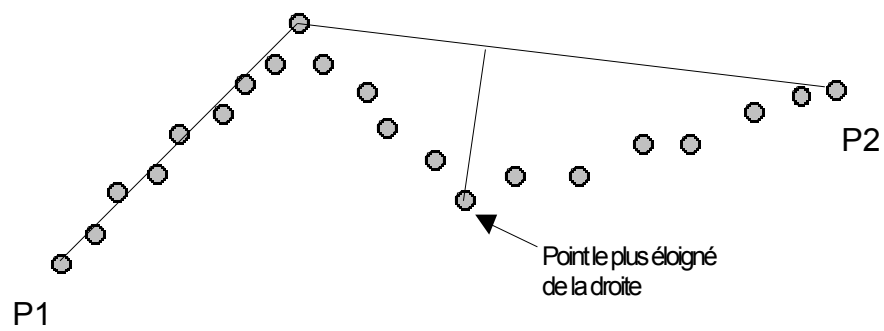
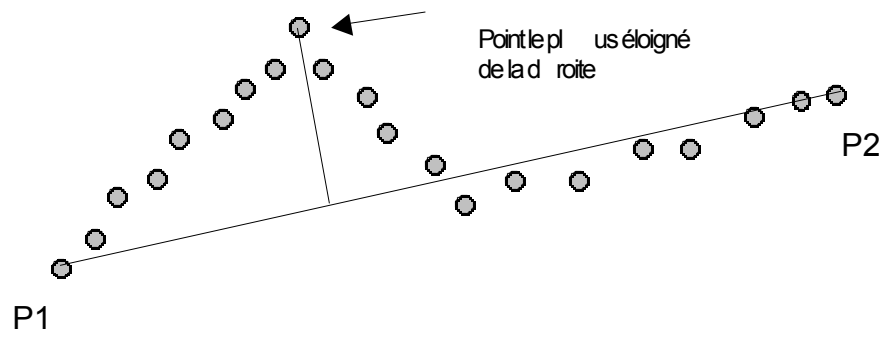
Le chaînage des contours est fait automatiquement par **MegaWave** lorsqu'on choisit les options appropriées. C'est de ces chaînes qu'il a été fait allusion ci-dessus. Une telle chaîne est une suite de points assez proches pour être considérés comme appartenant à un même contour. Une telle structure à l'état brut ne pourrait nous permettre de dire si une sous-chaîne forme un rectangle ou pas. Pour résoudre ce problème, on approxime les chaînes par des segments de droite en essayant de ramener sur une même ligne tous les points qui s'en écartent d'au plus une certaine distance qu'on appellera seuil. L'approximation par segments sera faite en utilisant le découpage récursif de la chaîne.

Algorithme de découpage récursif

1. On calcule l'équation de la droite passant par les deux extrémités d'une chaîne
2. Pour chaque point de la chaîne, on calcule la distance qui le sépare de la droite joignant les deux extrémités et on détermine le point le plus distant.
3. On compare la distance à un seuil
 - Si elle est inférieure à ce seuil, on arrête
 - Sinon, on coupe la chaîne au point le plus distant et on rappelle de manière récursive la fonction.

Soient P1 et P2 les deux extrémités de notre chaîne

Illustration du découpage récursif et calcul des distances



Si $P1$ a pour coordonnées (x_1, y_1) et $P2$ a pour coordonnées (x_2, y_2) ; la distance entre un point de coordonnées (x, y) à la droite décrite par $P1$ et $P2$ est : $d = |ax + by + c|$.

Où

$$a' = y_2 - y_1, \quad b' = x_2 - x_1 ;$$

$$a = a' * (a'^2 + b'^2)^{-1/2},$$

$$b = b' * (a'^2 + b'^2)^{-1/2} \text{ et}$$

$$c = -(ax_1 + by_1)$$

Description détaillée de l'algorithme de découpage récursif

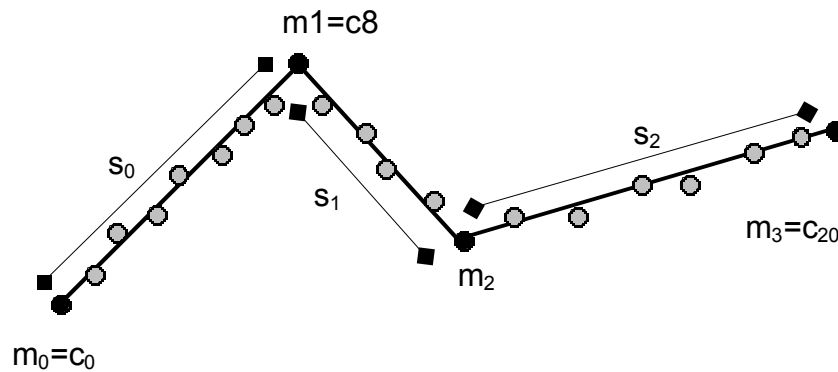
Soit $C = \{c_i, i=0, 1, \dots, n\}$ l'ensemble des points d'une chaîne ; $c_i = (x_i, y_i)$ étant le $i^{\text{ème}}$ point de la chaîne avec (x_i, y_i) ses coordonnées dans l'image. Le point c_i a, excepté lorsqu'il se trouve en bout de chaîne, 2 voisins qui sont c_{i-1} et c_{i+1}

Soit S une partition de C et M un sous-ensemble de C contenant les points de cassure correspondant à la partition P . On a

$$S = \{s_0, s_1, \dots, s_{N-1}\} \text{ et } M = \{m_0, m_1, \dots, m_N\}$$

Le segment s_k commence au point m_k et se termine au point m_{k+1} . Puisque M est un sous-ensemble de C , les éléments m de M appartiennent aussi à C donc chaque point m correspond à un point c : $m_k = c_{i_k}$. Un segment est ici défini comme étant un sous-ensemble de points de C tel que :

$$S_k = \{c_j / i_k < j < i_{k+1}, c_{i_k} = m_k, c_{i_{k+1}} = m_{k+1}\}$$



Exemple de segmentation

Soit une sous-chaîne s_k dont les extrémités sont m_k et m_{k+1} . Soit D une droite passant par m_k et m_{k+1} . Soit d_{ik} la distance entre le point c_{ik} et la droite D (c_{ik} se trouve entre m_k et m_{k+1}). Soit d_{max} la plus grande des distances d_{ik} . Nous appelons m' le point qui est situé à la distance d_{max} de la droite D .

$$d_{max} = \max(d_{ik}).$$

La chaîne est correctement approximée si la distance d_{max} est plus petite qu'un seuil que nous allons appeler seuil d'acceptation ou valeur d'acceptation. Dans le cas contraire on sépare la chaîne s_k en deux sous-chaînes s'_k et s''_k et on applique le même traitement. Avec s'_k qui est la sous-chaîne de s_k allant de m_k à m' et s''_k la sous-chaîne allant de m' à m_{k+1} .

Le seuil défini ci-dessus est généralement choisi à partir d'observations et d'expérimentations. Il dépend de plusieurs paramètres donc la résolution de l'image.

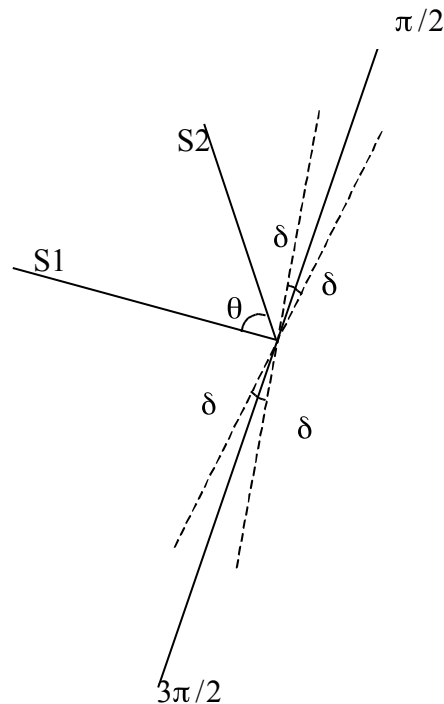
- **Recherche des couples de segments connexes à peu près perpendiculaires**

Dans cette partie, on étiquette chaque nœud m_k par un code qui est en rapport avec l'angle entre les deux segments qui lui sont adjacents.

Choix du codage pour l'étiquetage des points liant deux segments

L'opération d'étiquetage des points ne se fait que pour les points qui sont entre deux segments, c'est à dire des points qui ne sont pas aux extrémités de la chaîne. Nous utilisons 4 valeurs pour réaliser ce codage : -2, -1, 1 et 2. Ces étiquettes renseignent à la fois sur le signe de l'angle entre les segments qui concourent au point étiqueté et permet de dire si oui ou non on peut supposer que ces deux segments sont perpendiculaires.

Les valeurs négatives (-2 et -1) signifient que l'angle entre les deux segments est dans l'intervalle $[\pi, 2\pi[$ et les valeurs positives expriment le fait qu'il est dans l'intervalle $[0, \pi[$. Les valeurs -1 et 1 indiquent que les segment qui concourent à ce point sont considérés comme formant un angle droit tandis que les valeurs -2 et 2 indiquent le contraire.



Soit θ l'angle entre le segment S1 et le segment S2, δ la marge sur $\pi/2$ et $3\pi/2$ pour laquelle on continue à considérer qu'un angle est droit. On a :

$$1 \text{ si } \theta \in [\pi/2 - \delta, \pi/2 + \delta] = I_1$$

$$-1 \text{ si } \theta \in [3\pi/2 - \delta, 3\pi/2 + \delta] = I_2$$

$$2 \text{ si } \theta \in [0, \pi[\text{ et } \theta \notin I_1$$

$$-2 \text{ si } \theta \in [\pi, 2\pi[\text{ et } \theta \notin I_2$$

- **Regroupement de ces couples de segments pour détecter les rizières et les zones de rizières**

Après l'opération de codage, cette étape consiste au regroupement des segments pour constituer des rizières. Pour savoir si l'on a un ou plusieurs rectangles pour une suite de segments donnée, il suffit de chercher si l'on a trois valeurs d'étiquettes successives qui sont soit tous à 1 soit tous à -1 et que le premier point de la suite des 5 coïncide avec le cinquième. Chaque rectangle qui sortira de là sera une rizière.

L'algorithme de découpage en C++

```

/* p1, p2 sont des pointeurs sur
les deux extrémités d'une chaîne et
e le seuil à partir duquel on
considère que qu'un point fait
partie d'un segment */

void decoupeRec(p1, p2, e)
Point_curve p1, p2; /* Point_curve est un type de
Megawave2*/

float e;
{
    Point_curve p, q;
    float d;
    plus_eloigne(p1, p2, p, &d); /*On recherche ici le point p et la
distance qui le sépare de la droite
décrite par p1 et p2 */

    if (d <= e)
        segment(p1,p2); /* Si cette distance est plus petite
que le seuil on constitue un segment
en éliminant les points
intermédiaires*/

    else /* Sinon, on découpe notre chaîne en
deux parties au point p et applique
le même algorithme sur chacune des
parties */

        {
            decoupeRec(p1, p, e);
            decoupeRec(p, p2, e);
        }
}

void codage(p1, p2)
Point_curves p1, p2;
{

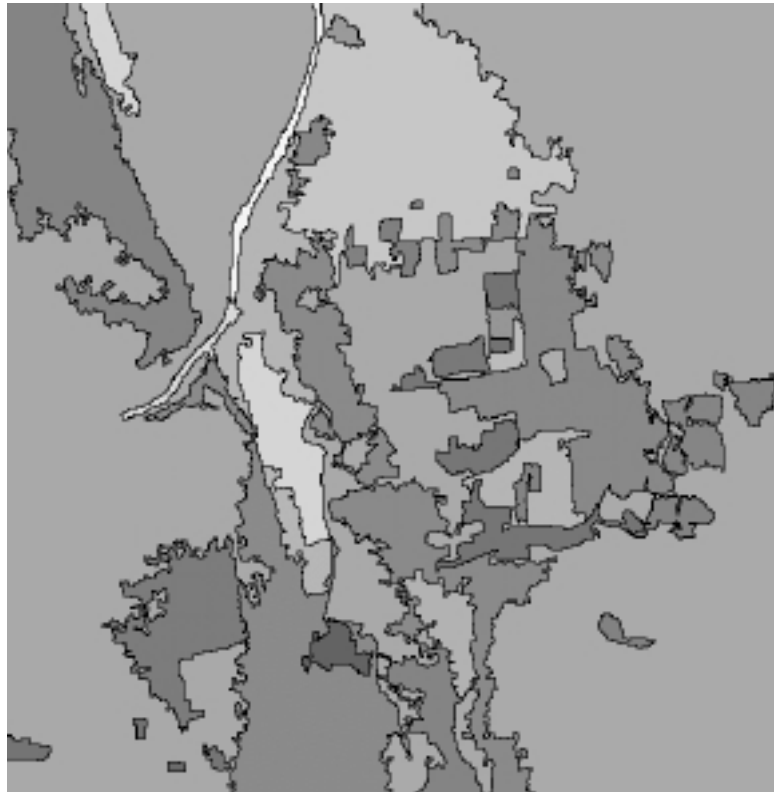
}

```

Les images qui suivent illustrent cette approche.

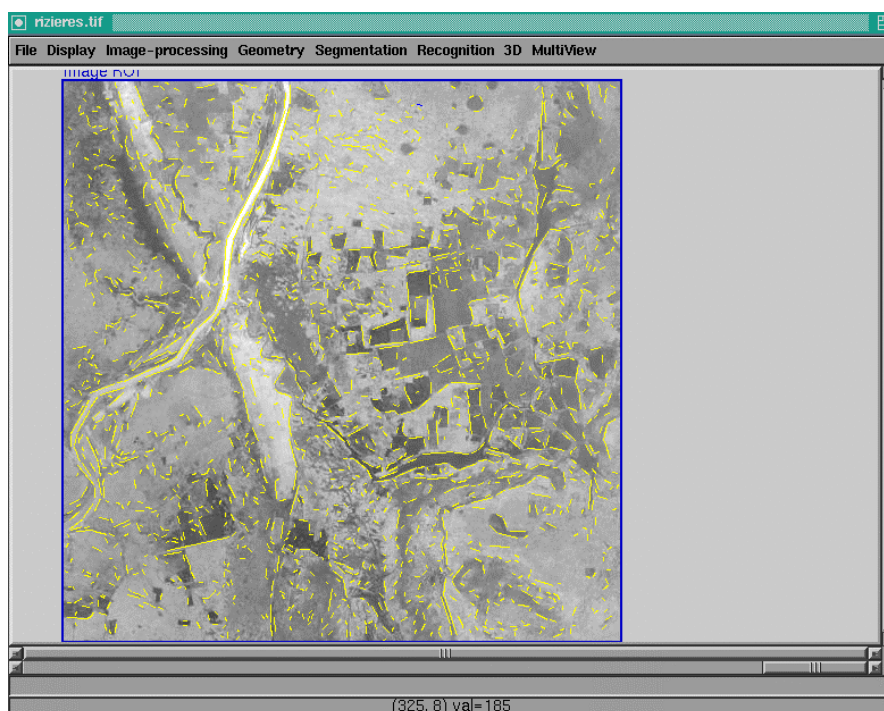


Image originale de rizières (Delta Central du Niger)



Résultat obtenu par superposition des contours sur les régions

L'image ci-dessous est issue de la superposition des régions et des contours sur l'image originale du Delta. Elle permet de mieux différencier les régions extraites et d'apprécier les formes des régions obtenues.



Chaînage et approximation polygonale des contours

2.3. Méthode "régions"

L'approche "régions" (voir deliverable D11) comporte les étapes suivantes :

- *Segmentation de l'image en régions homogènes au sens des niveaux de gris*

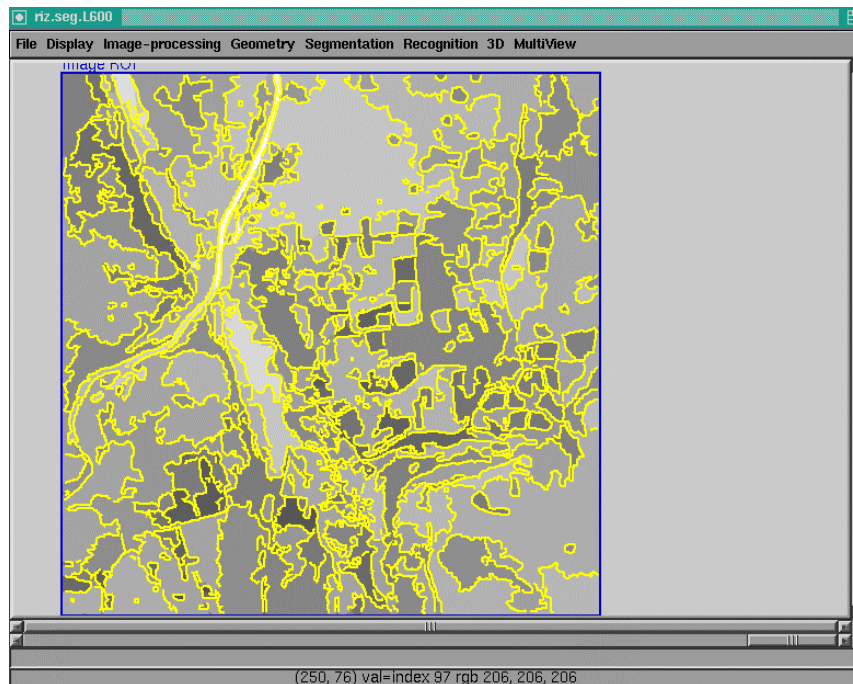
Avec Megawave2 cette partie est pratiquement la même que la précédente car le module **segct** qui traite de la segmentation des contours permet aussi, selon les résultats sélectionnés, de produire une image des régions au sens des niveaux de gris. La méthode utilisée pour la détection des contours et des régions est celle de Mumford-Shah. Nous nous contentons ici de rappeler succinctement les différentes méthodes de détection de contours et de régions, car ces méthodes sont décrites en détail dans le D11.

- *Sélection des régions à peu près rectangulaires :*
 - *Calcul des axes principaux d'inertie des régions*
 - *Détermination du rectangle d'aire minimale englobant la région dont les axes sont les axes principaux d'inertie*
 - *Calcul d'un indice de "rectangularité" par le rapport entre l'aire de la région et celle du rectangle englobant*
 - *Sélection de l'ensemble des régions dont l'indice de "rectangularité" est suffisamment élevé.*
- *Regroupement des régions rectangulaires pour déterminer les zones de rizières.*

Les images qui suivent illustrent cette approche en cours d'implémentation.



Résultat de la segmentation en région (méthode de Mumford-Shah ; chaque région est coloriée avec son niveau de gris moyen) ; les zones de régions rectangulaires correspondent aux zones de rizières.



Contours des régions.

2.4. Conclusion

L'extraction des rizières et zones de rizières est relativement aisée à partir des résultats de la segmentation (contours ou régions). En effet, les rizières correspondent à des zones de caractéristiques géométriques (rectangles) et photométriques (variance faible) spécifiques. Les résultats obtenus sont plus ou moins satisfaisants pour le moment mais sont très prometteurs.

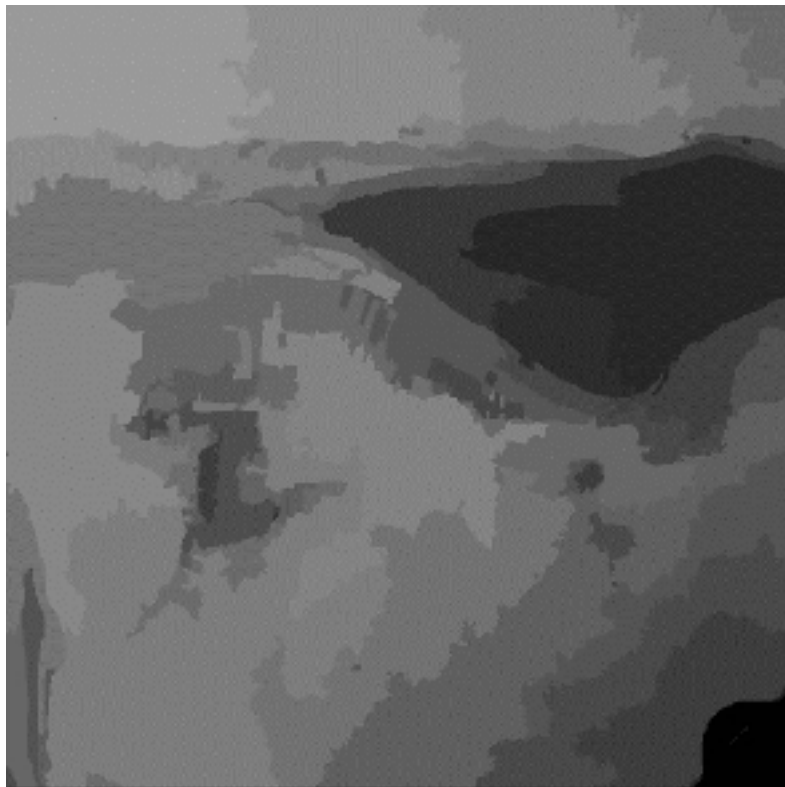
3. Extraction des zones inondées

Dans les images les zones inondées correspondent à des zones homogènes au sens des niveaux de gris sans forme géométrique spécifique. Ainsi leur identification de manière complètement automatique est un problème difficile à partir d'images optiques monochromatiques (elle pourrait être plus facile à partir d'image radar).

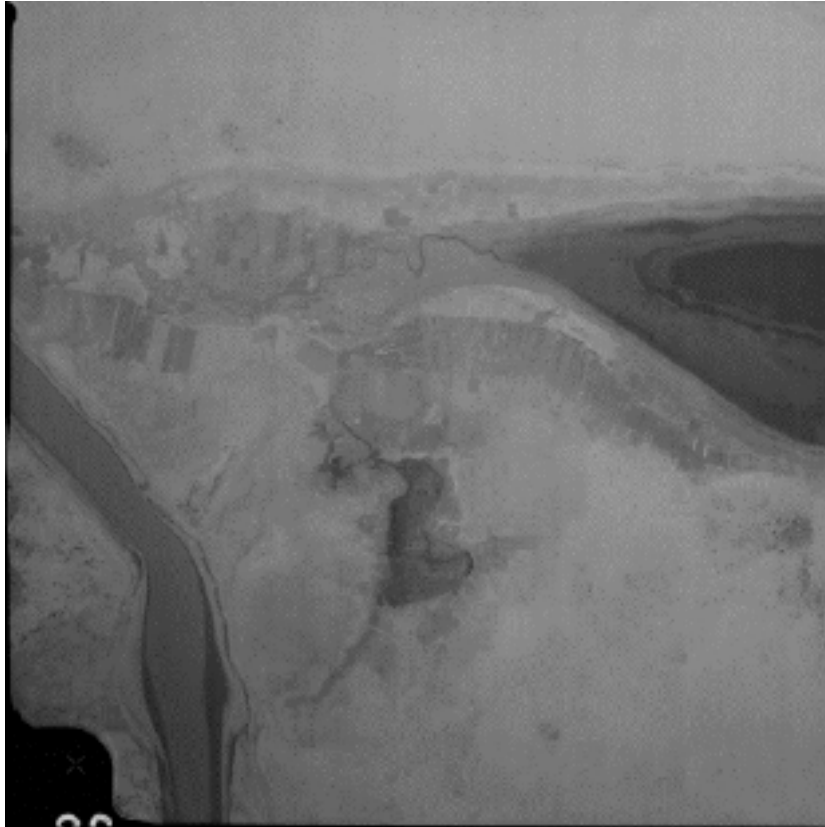
Ce problème est moins difficile dans les cas où la plage des niveaux de gris des zones inondées est spécifique. L'approche que nous utilisons nécessite une interaction avec un opérateur qui sélectionne les régions correspondant aux zones inondées à partir de l'image segmentée. Les images ci-après illustrent la pertinence de cette approche.



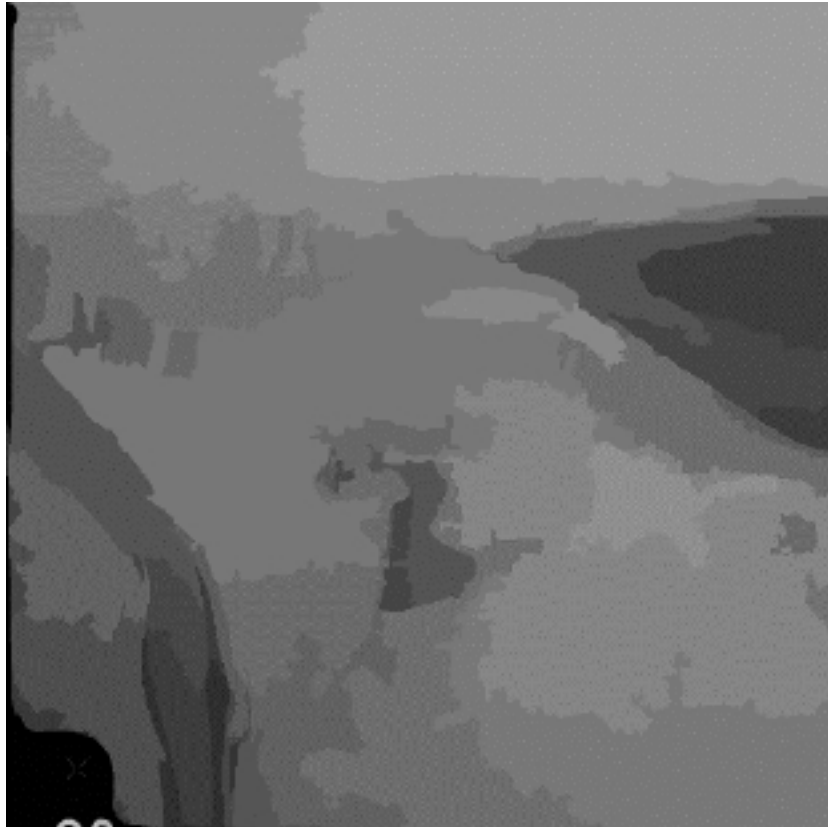
Zone inondée du Delta Central du Niger (1)



Résultat de la segmentation en régions (algorithme de Mumford-Shah)



Zone inondée du Delta Central du Niger (2)



Résultat de la segmentation en régions (algorithme de Mumford-Shah)

4. Extraction des rivières et des routes

4.1. Introduction

Dans les images aériennes et satellitales, les routes, fleuves et voies ferrées correspondent à des structures dites « en ruban ». Nous avons introduit un nouvel algorithme de détection automatique de ces structures décrit en détail dans les articles en référence et dans la thèse de Véronique Prinet (Université d'Orsay, 1998) dont le travail a été en partie financé par le projet SIMES. Nous illustrons ci-après les étapes de cet algorithme qui, contrairement à la plupart des approches existantes, ne prend en compte aucune information a priori et s'adapte à tout type de structure en ruban (largeur, contraste, forme, ...). Mais avant de décrire l'implémentation de cette méthode, nous allons citer ici quelques méthodes actuelles qui permettent de réaliser l'extraction des structures en ruban dans les images. Il s'agit en fait de :

- la modélisation probabiliste des routes, c'est une modélisation géométrique des routes appliquée aux réseaux épais.
- les bancs de filtres gaussiens pour la détection des réseaux linéiques (exemple des vaisseaux sanguins en imagerie médicale)
- les réseaux de neurones entraînés pour détecter les routes dans les images LANDSAT
- la programmation dynamique
- la théorie de l'information : à partir d'un point et d'une direction donnée, l'algorithme effectue des tests sur la radiométrie des pixels dans la direction de la route et dans la direction orthogonale à la route. Les pixels sélectionnés comme appartenant à la route sont ceux qui minimisent une fonction d'entropie.

4.2. Méthode « géométrie différentielle »

Le principe de cette nouvelle approche consiste à extraire les lignes de crête de la surface image définie par : $(x, y, I(x, y))$; I étant la fonction des niveaux de gris. Ces lignes de crête correspondent aux structures en ruban et sont déterminées par calcul des extrema locaux des courbures maximales dans la direction maximale de courbure. Ce calcul est réalisé grâce à des techniques avancées (filtrage récursif d'ordre 3 et 4) de filtrage linéaire permettant d'obtenir les dérivées d'ordre 1 et 2 du signal image. Les maxima locaux des courbures, caractérisés par les zéros des fonctions dérivées d'ordre 1, 2 et 3 de la fonction image, sont déterminés par interpolation locale à partir des courbures et des directions principales.

Une approche multi-résolution, basée sur le calcul des dérivées de l'image avec des facteurs différents de lissage, permet d'être robuste au bruit et de s'adapter à toutes les largeurs des structures. Une description détaillée de cette méthode peut être trouvée dans les références données après la conclusion de cette partie, notamment dans la publication jointe en annexe.

Les axes des structures linéaires sont localisés en utilisant les caractéristiques de la dérivée d'ordre 3. On n'obtient qu'une partie de l'information sur l'objet. Cette information est complétée dans l'étape suivante qui consiste à reconstruire le réseau à partir des éléments locaux obtenus à l'étape précédente et consignés dans un graphe. Avec cette structure de graphe, et en utilisant la théorie des graphes, essaie de mettre bout à bout deux ou plusieurs

arêtes entre elles pour former une nouvelle structure linéaire. Pour deux arêtes prises au hasard, on calcule leur intersection et on obtient un point C . On calcule ensuite la distance entre ce point et chacun des segments. On crée un nouveau sommet si C appartient à l'un des segments et si la distance entre ce point et l'autre segment est plus petite qu'un certain seuil qu'on fixe au moment de lancer l'opération. La troisième étape du processus est une étape de classification car jusqu'ici on a seulement détecté les structures linéaires sans chercher à quoi elles correspondent exactement.

Nous nous basons ici sur les observations du monde réel pour caractériser et différencier les routes des cours d'eau. On se base sur la nature des points d'attache C qui ont été extraits et également sur les propriétés radiométriques et géométriques des lignes autour de chacun de ces points. Cette opération est réalisée lors de la construction du graphe. Nous définissons quatre types de points de liaison C :

- **« T » ou « + » si C résulte de l'intersection de deux lignes perpendiculaires**
- **« X » ou « Y » sinon**
- **« T » et « Y » sont les jonctions entre l'extrémité d'une ligne et n'importe quel point d'une autre ligne.**
- **« X » et « + », C appartient aux deux lignes.**

Nous considérons ici deux classes de réseaux : les rivières et les routes. Nous avons dit plus haut que c'est le type de jonction qui va déterminer la classe de réseau auquel les lignes appartiennent. Lorsqu'une jonction a été identifiée comme appartenant à une classe, toutes lignes qui lui seront connectées devront appartenir à la même classe.

Les figures ci-après illustrent les principes et les résultats de cet algorithme.

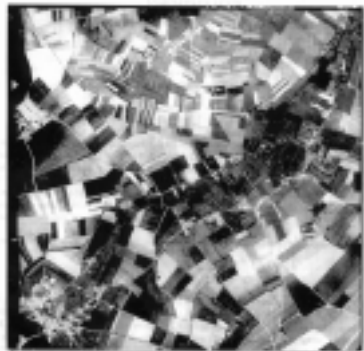


Figure 7 - (a) Image aérienne de la région de Long la Campagne (niveau 1000) (b) Détection des structures linéaires présentes dans un espace urbain

Madelaine

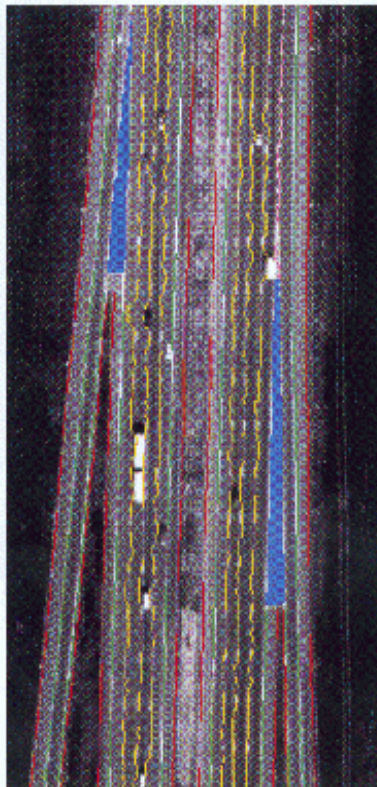


Fig. 4.22 Results on large scale of detection of road side or components of a 3-pm network in dense area and detection of road width by the roadwidth estimator

LEGEND	
█	Road side
█	Lane line
█	Edge line
█	White triangle
█	Track line

De gurst

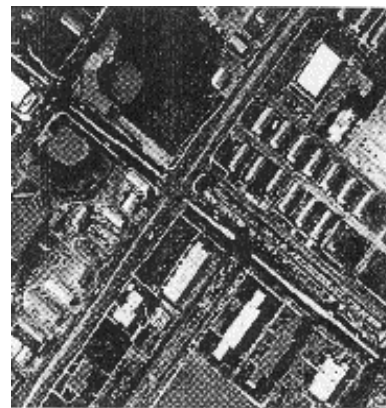


Figure 8: Road Image number 2 called "Road2" taken from the Radius program.

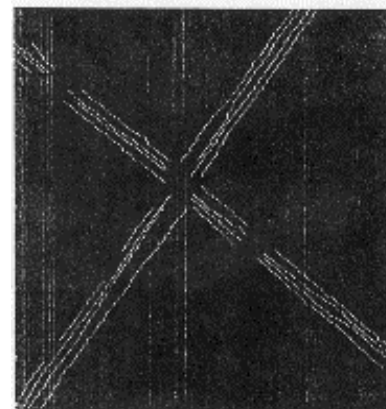


Figure 9: The recognized and estimated roads and barriers of our road finder, in image "Road2".

Cooper

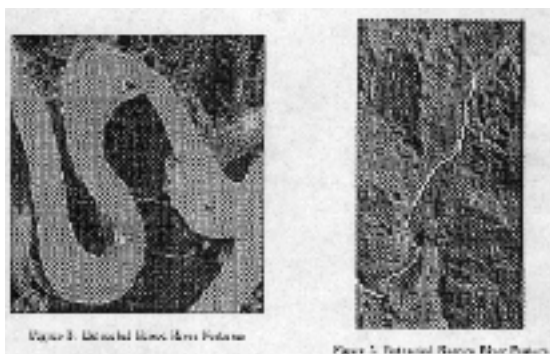


Figure 3: Roadwidth Estimator (Left) Posture

Figure 5: Roadwidth Estimator (Right) Posture

Dobie

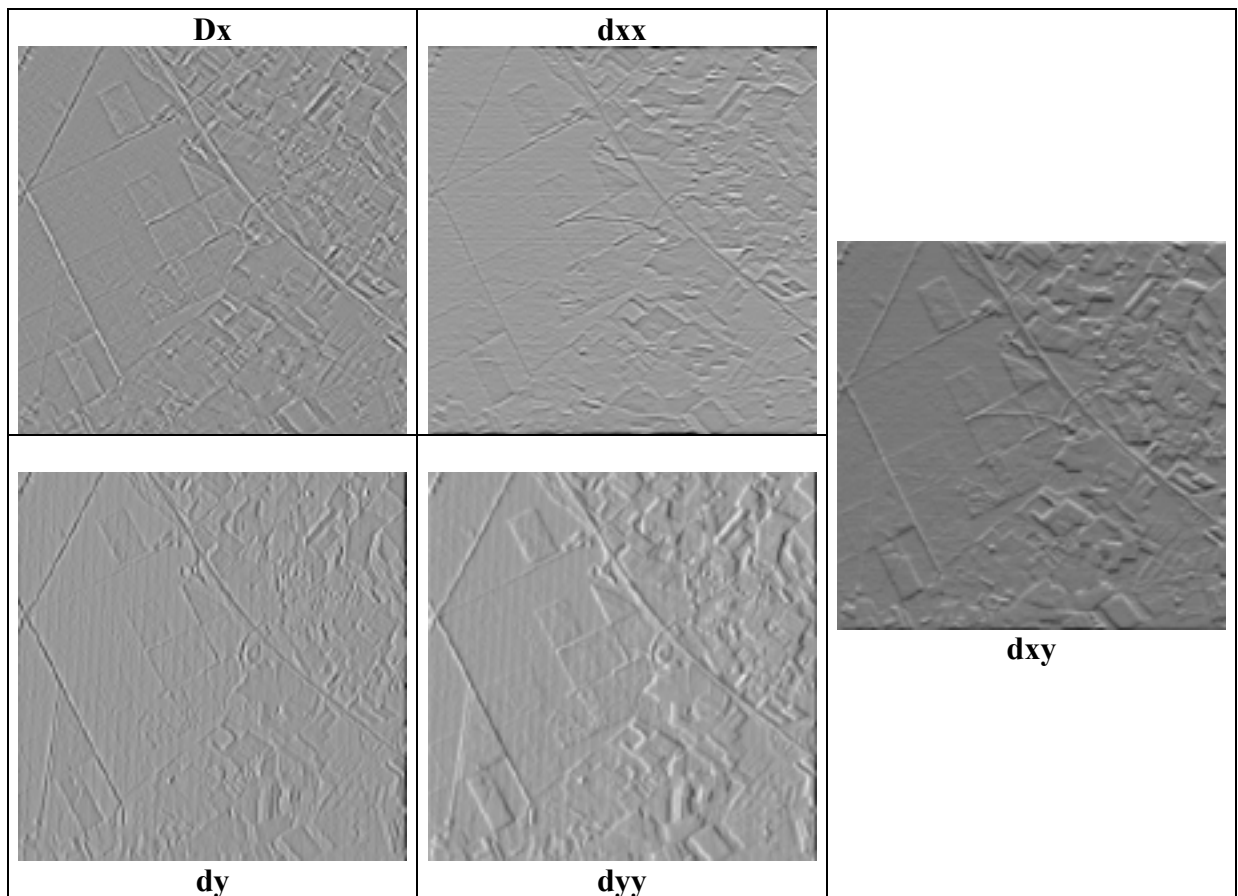
Les noms sous les images sont ceux des personnes qui ont fait tourner leurs algorithmes sur les images en question.

Comme il a été mentionné plus haut dans ce document, nous utilisons à certains moments des images autres que celles du Delta pour montrer que les algorithmes que nous proposons sont aussi, et parfois plus, efficaces sur des régions fortement urbanisées.



Image originale

Cette image représente une zone d'habitations traversée par des routes. On peut en extraire tant des structures linéiques (ici, des routes) que des zones rectangulaires (ici, des bâtiments ou des parcelles). Ci-dessous est présentée une suite de traitements appliqués sur cette image, avec les résultats, étape par étape.



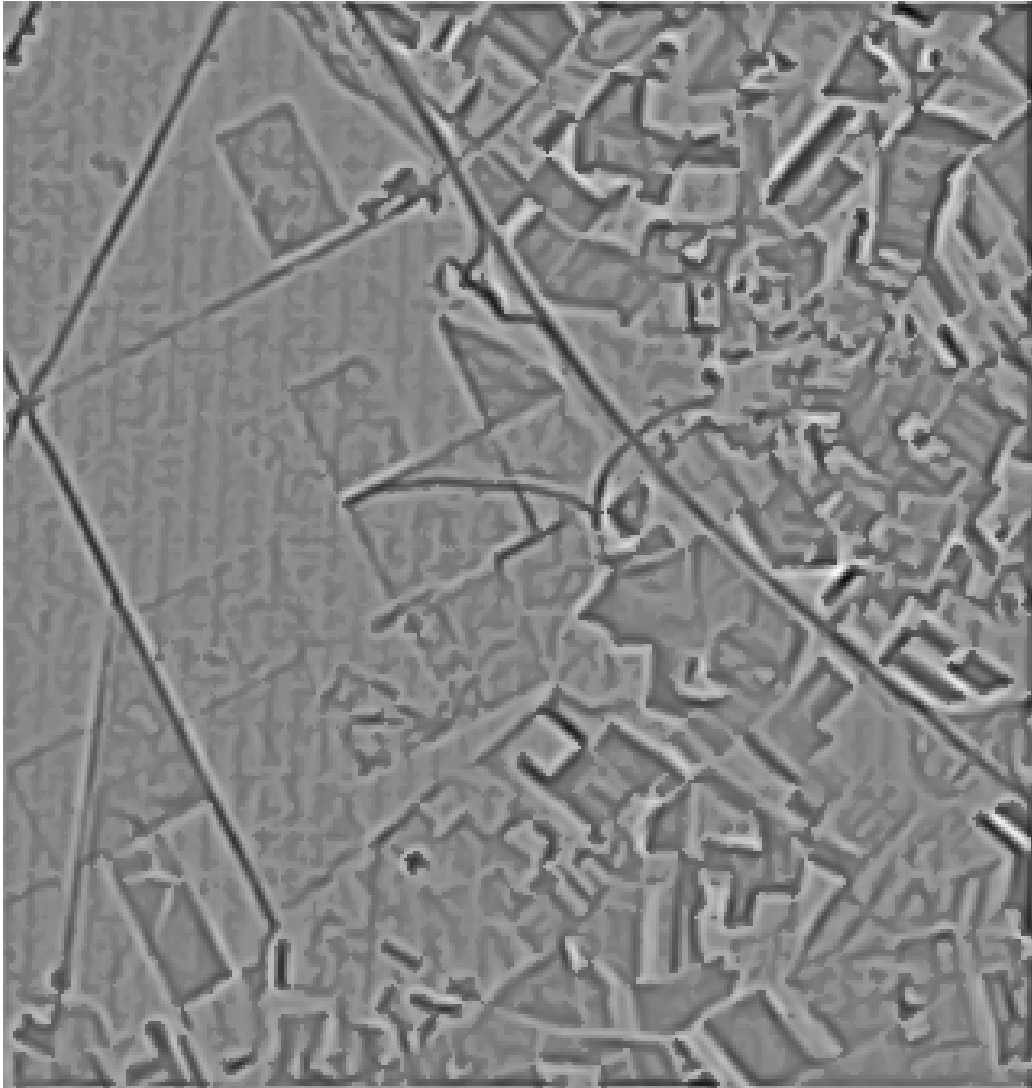


Image des courbures maximales

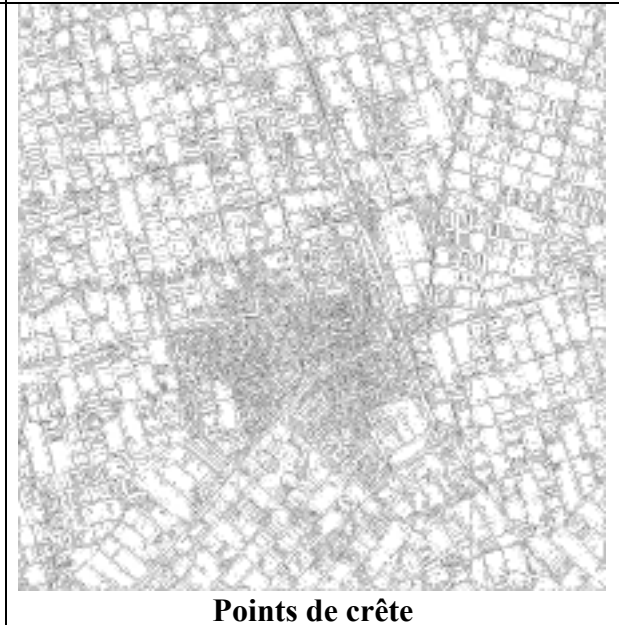
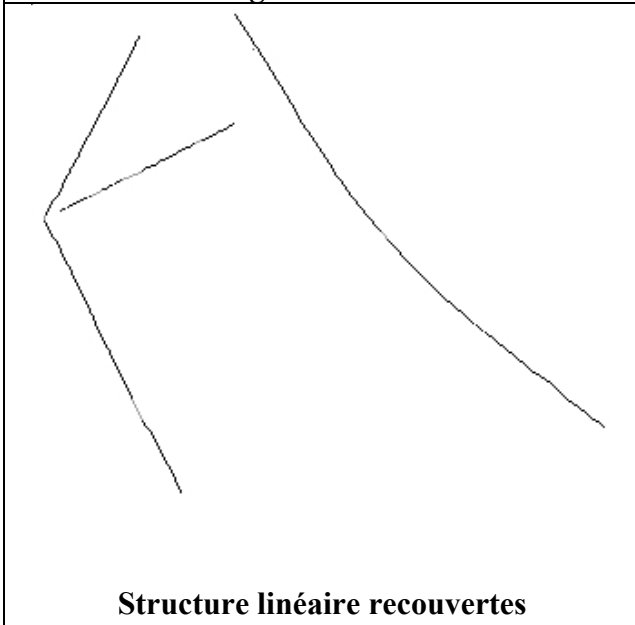
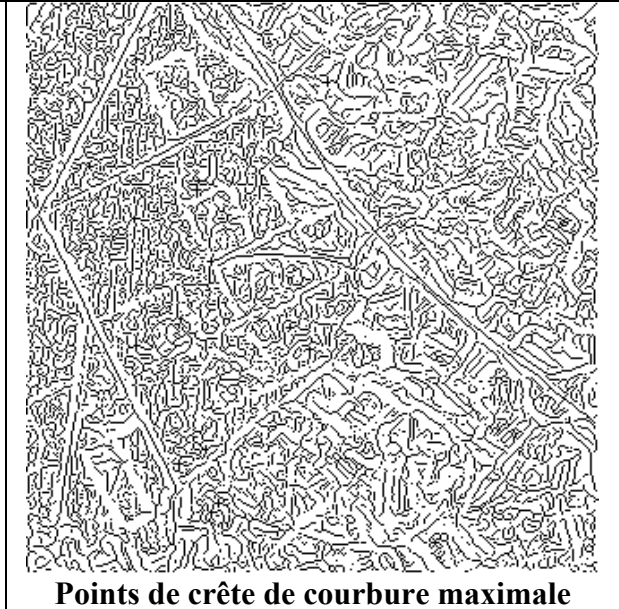
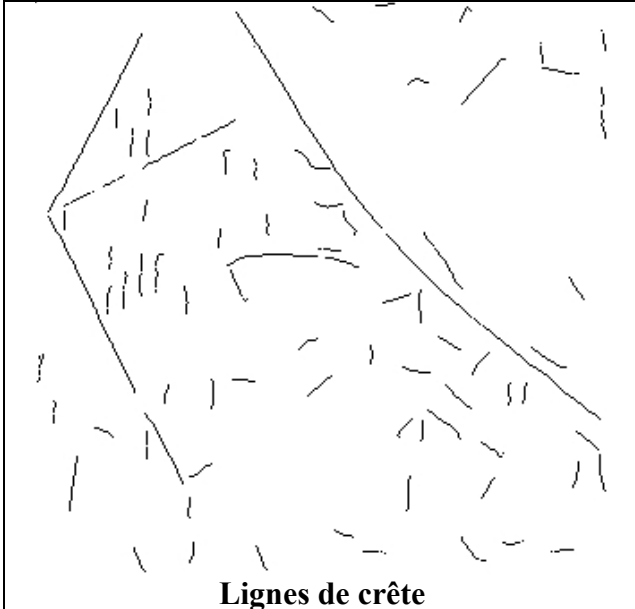
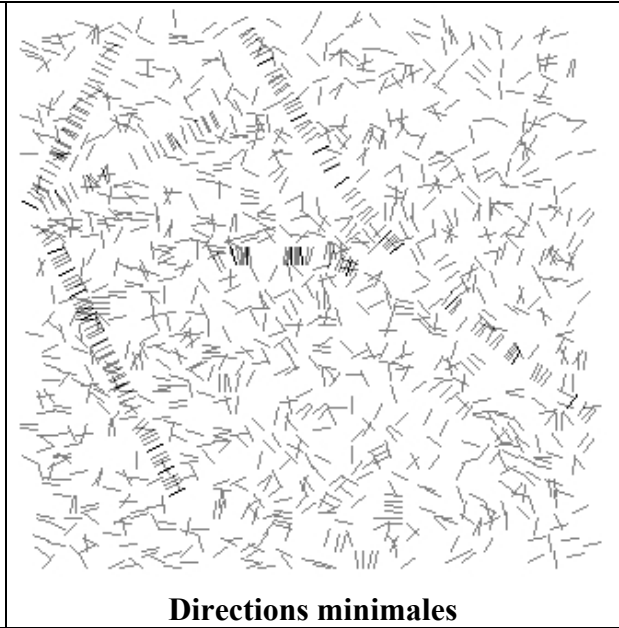
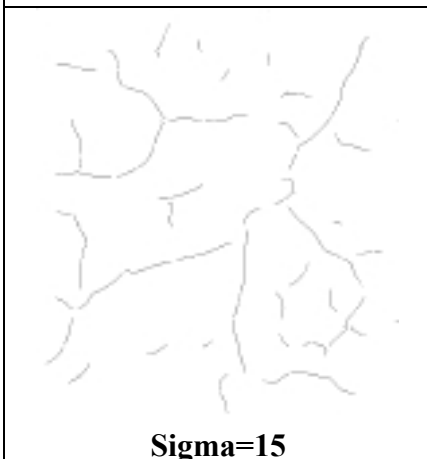




Image originale



Sigma=15



Sigma=20



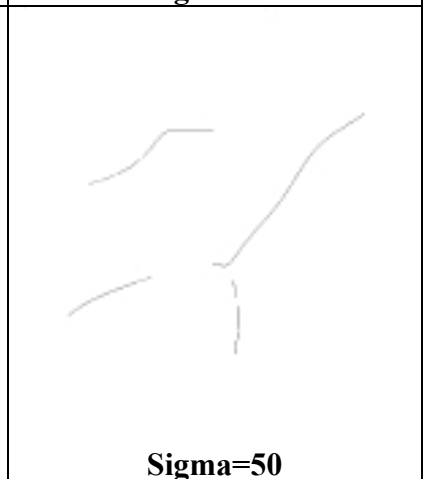
Sigma=25



Sigma=30



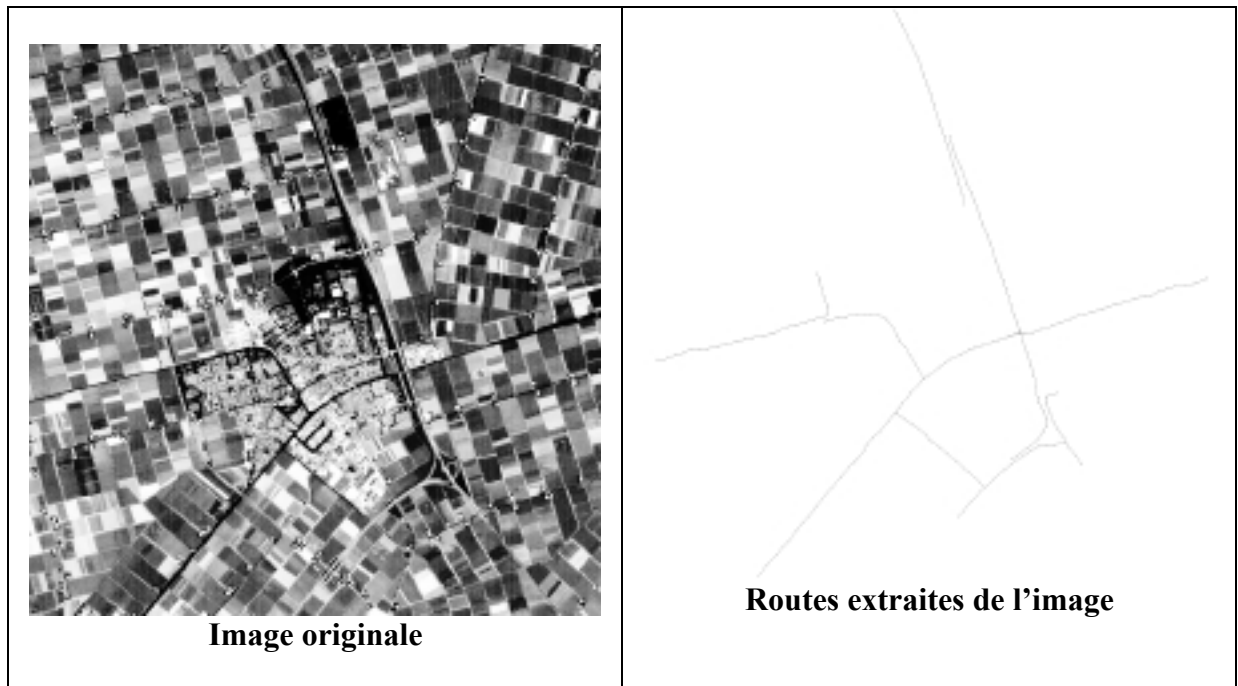
Sigma=40



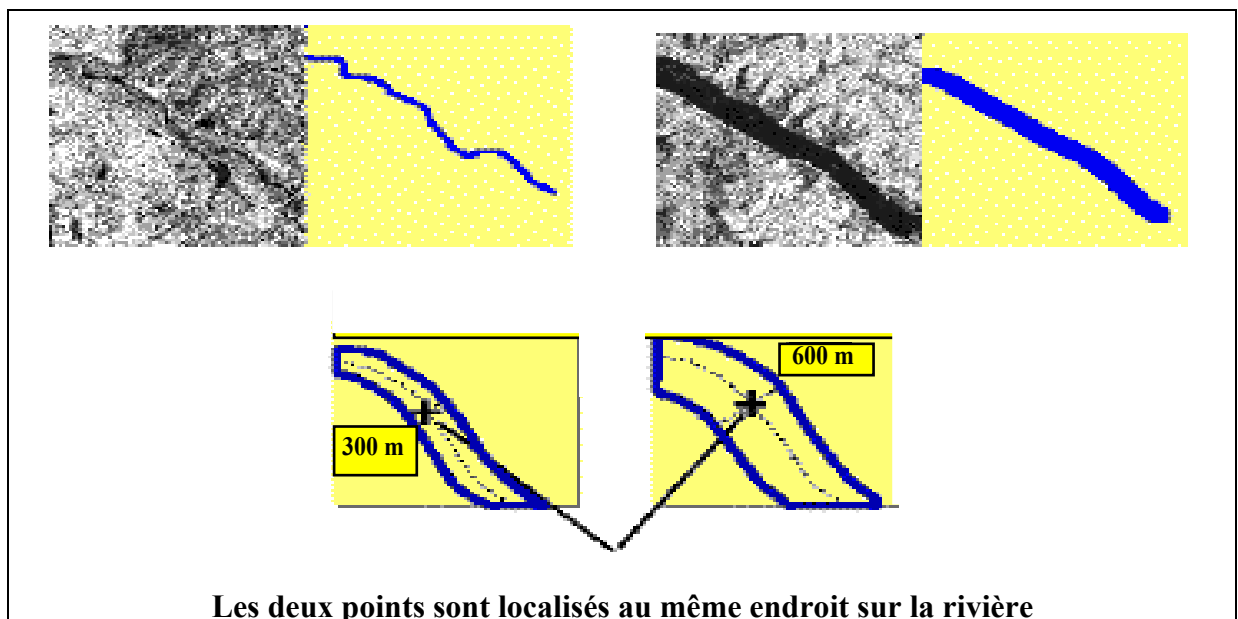
Sigma=50

Sigma est l'écart type du filtre qui balaie l'image lors de l'opération de détection de contours.

Autre exemple :



Les images suivantes montrent la robustesse de l'algorithme. Nous avons ici une même rivière en temps de crue et en période d'étiage. On arrive à détecter dans ces deux cas extrêmes le lit de la rivière et sa largeur.



Les deux points sont localisés au même endroit sur la rivière

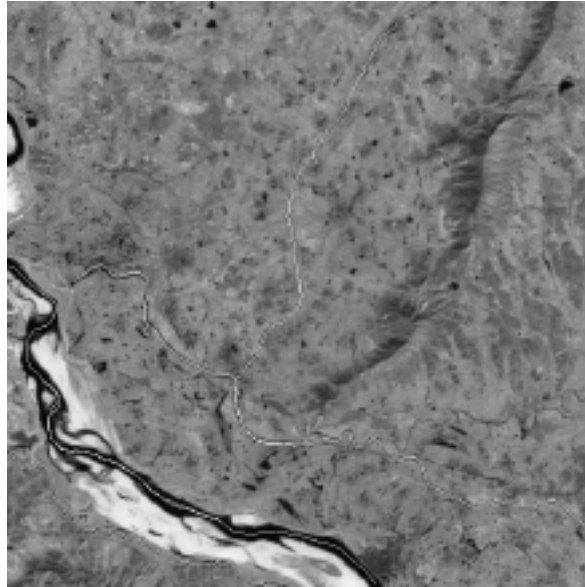
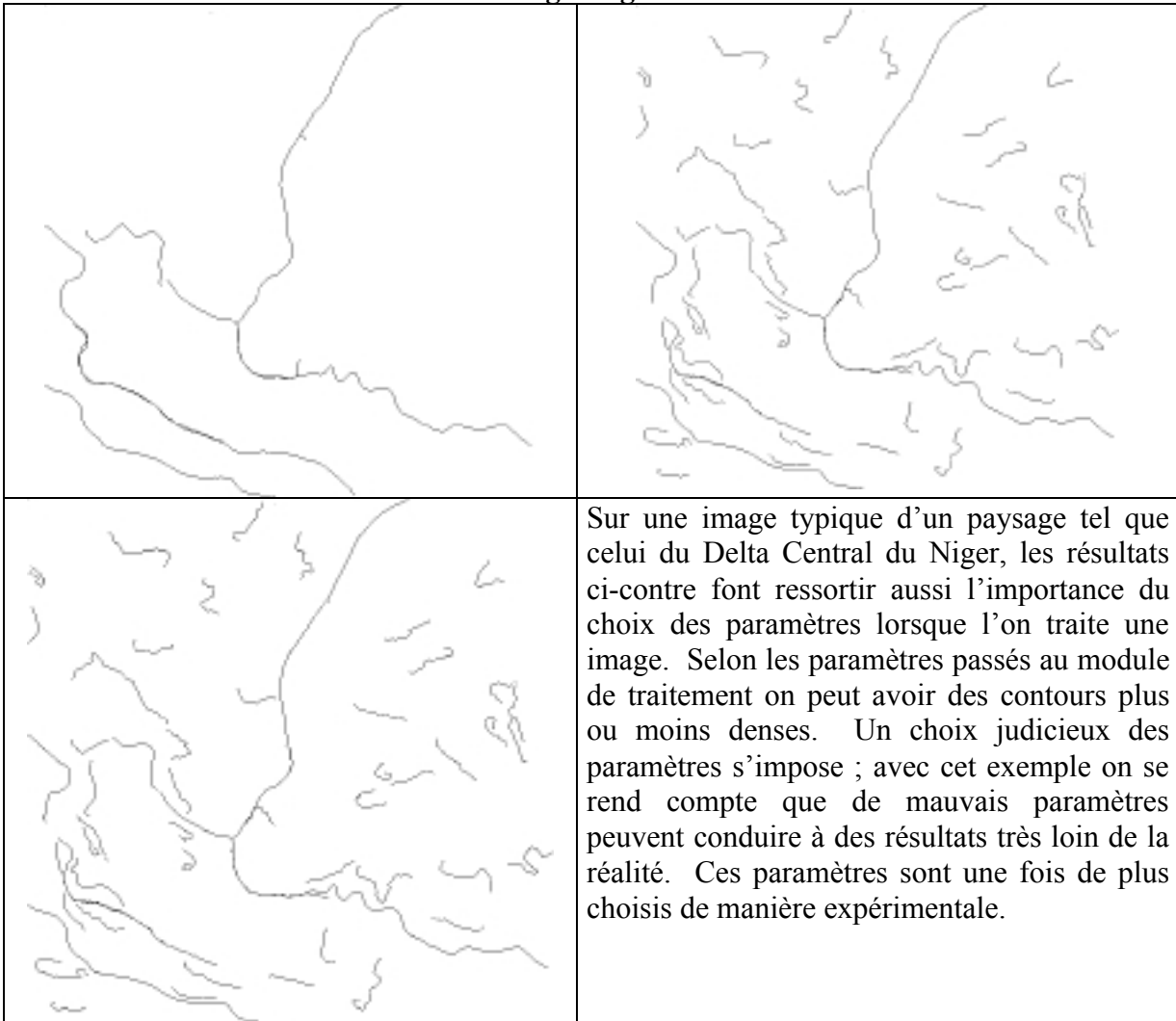
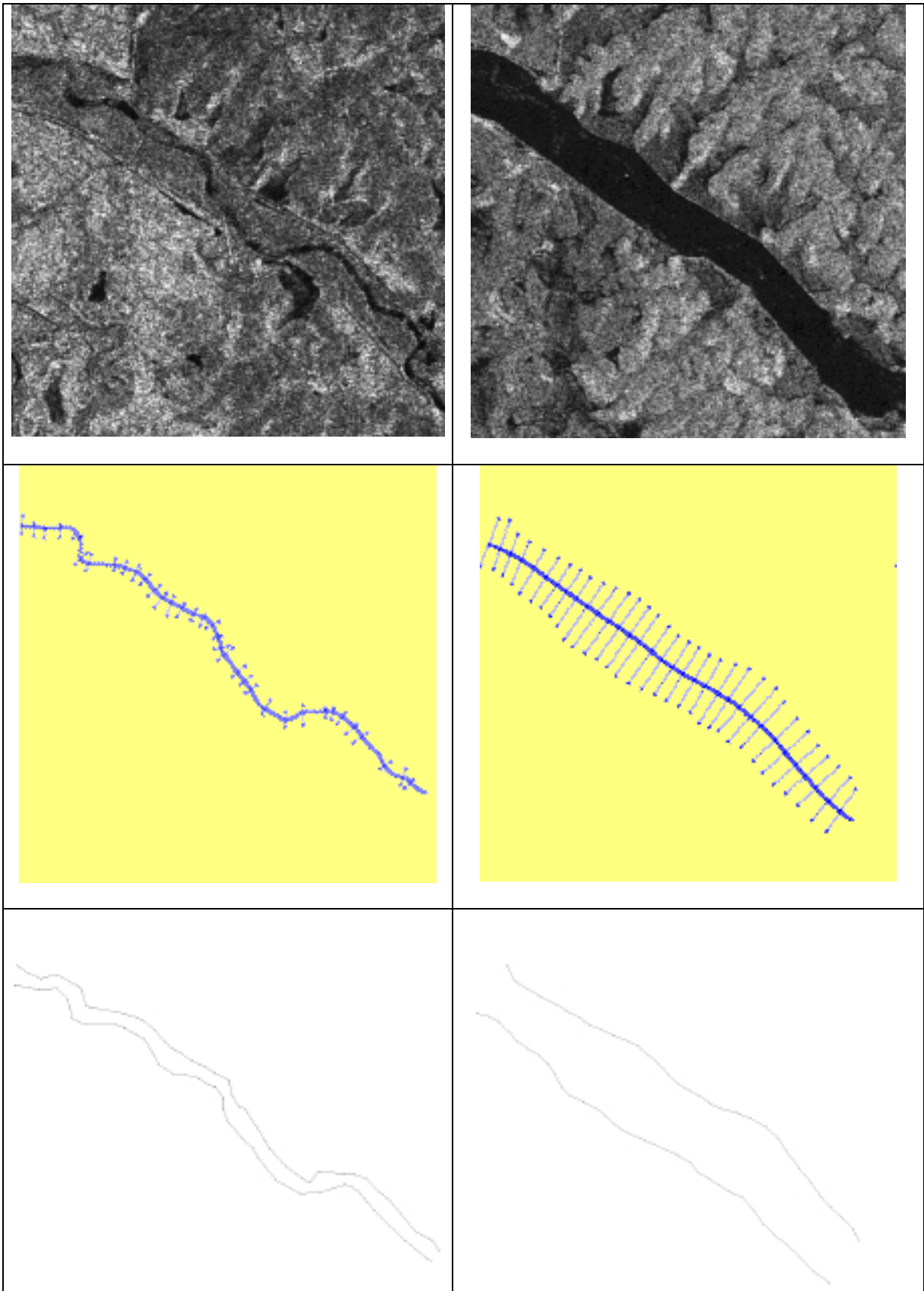


Image originale





Nous avons ci-dessus une illustration des étapes de l'algorithme où ressort d'abord l'axe de la rivière, dont le lit est ensuite reconstruit en utilisant les caractéristiques radiométriques.

5. Conclusion

Les résultats obtenus jusqu'ici par les méthodes d'extraction de paramètres sélectionnées nous permettent de penser qu'on peut réaliser un outil fiable qui donne aux utilisateurs une plus grande facilité d'accès aux informations sur une image. Ces méthodes sont encore à parfaire : surtout du point de vue de la programmation où certains résultats qui semblent très bons en théorie ne donnent pas entière satisfaction.

Annexe : publication jointe

L'annexe jointe contient une publication de Véronique Prinnet dont la thèse a été en partie financée par le projet SIMES. Cet article a pour titre « *Communication network recognition from SPOT images* ».

Références

- O. Monga, N. Armande, P. Montesinos. Thin nets and crest lines : application to Satellite data and Medical Images. *CVGIP:Image Understanding Volume 66, Number 1 Juillet 1997*, aussi rapport de recherche INRIA 1995 ; une version réduite est dans *Lecture Notes in Computer Science, Springer Verlag, IPMI'95*
- N. Armande, P. Montesinos, O. Monga, Guy Vaysseix. Thin Nets Extraction using a Multi-scale Approach *CVGIP:Image Understanding Volume 73, Number 2 February 1999*
- V. Prinnet, ZX Zhou, S.D. Ma, O. Monga "Communication network recognition from SPOT images" *International Geoscience And Remote Sensing Symposium*, Juillet 98, Seattle (USA)
- Véronique Prinnet, Ma Songde, and Olivier Monga "Classification and representation of networks from satellite images", *International Conference on Image Analysis and Processing*, Oct. 99, Venise (Italie)
- Véronique Prinnet, MA Songde, and Olivier Monga "Scale selection for curvilinear structures detection : application to remote-sensed images", *Asian Conference on Computer Vision*, Janvier 2000, Taipei (Taiwan)
- Radu Stoïca, "Processus ponctuels pour l'extraction des réseaux linéiques dans les images satellitaires et aériennes", Thèse de l'université de Nice-Sophia Antipolis, fev 2001
- O. Monga & Y. Poncet. « Techniques de vision par ordinateur appliquées à des images aériennes en zone inondable ». A paraître en 2001 dans *Zones Inondables Tropicales*, Orange D., Arfi R., Kuper M., Morand P., Poncet Y. éd. sc., Actes du Séminaire International GIRN-ZIT, Bamako, Juin 2000.