

Extraction de routes dans des images satellite

Alexandre Havard n° 8010

Seuillage

Initialisation
Seuil naïf
Sélection à deux
seuils

Composantes connexes

Recherche des
composantes
Filtrage

Améliorations

Ajustement des
seuils
Remplissage

Extractions effectives

Noirmoutier
Autre réseau
routier
Approfondissement
possible

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Comment extraire informatiquement le réseau routier d'une image satellite ?

- 1 **Seuillage**
 - Initialisation
 - Seuil naïf
 - Sélection à deux seuils
- 2 **Composantes connexes**
 - Recherche des composantes
 - Filtrage
- 3 **Améliorations**
 - Ajustement des seuils
 - Remplissage
- 4 **Extractions effectives**
 - Noirmoutier
 - Autre réseau routier
 - Approfondissement possible

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

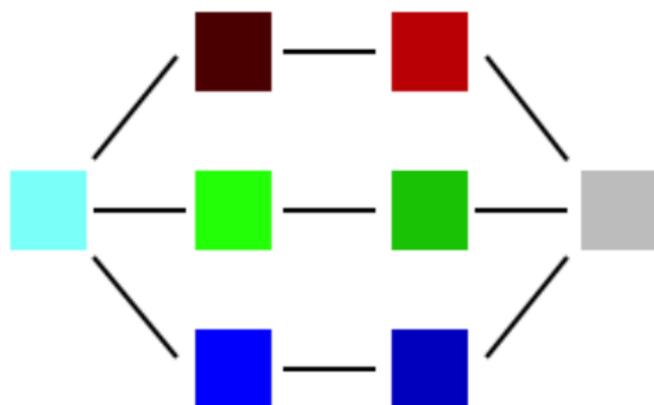
Autre réseau
routier

Approfondissement
possible

Seuillage

Passage en noir et blanc

$$G = \frac{R + V + B}{3}$$



- GREY : $G = 0,78$

Figure – Pixel avec $R=0.36$; $V=1$; $B=0.99$

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Seuil naïf

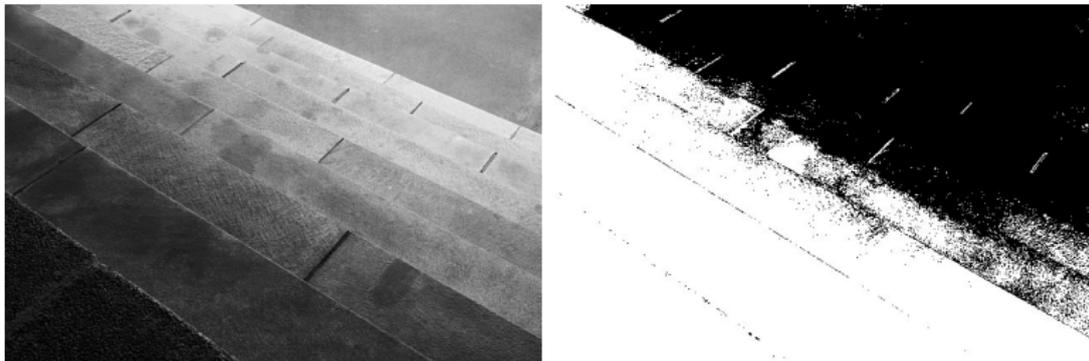


Figure – Seuil 0.5

Sélection à deux seuils

Seuillage

Initialisation

Seuil naïf

**Sélection à deux
seuils**

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

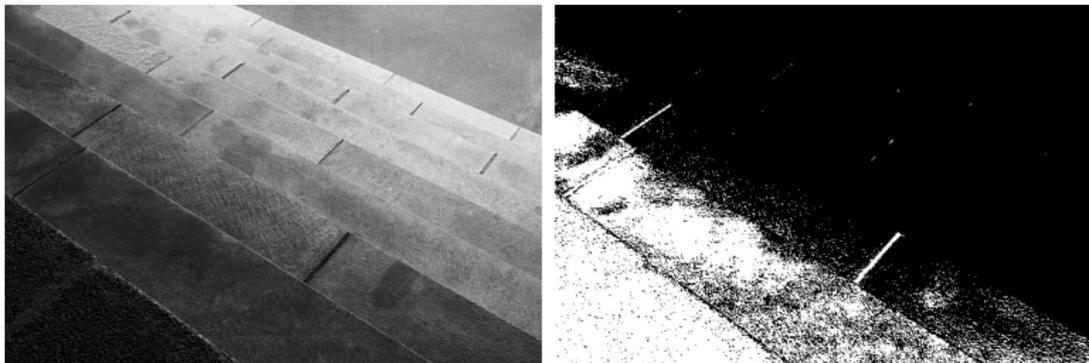


Figure – Seuils 0.2 / 0.5

Sélection à deux seuils

Seuillage

Initialisation

Seuil naïf

**Sélection à deux
seuils**

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

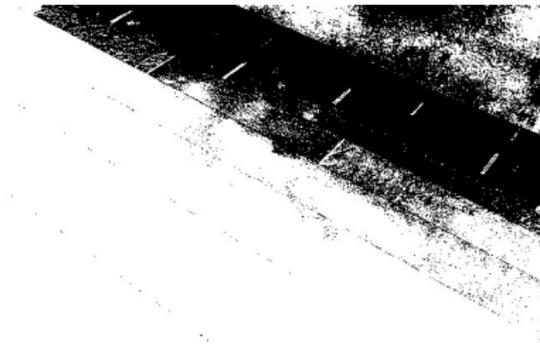
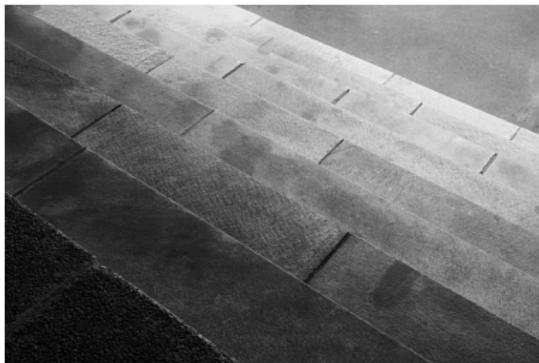


Figure – Seuils 0.6 / 0.7

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Composantes connexes

Explication connexité



- Composante connexe



- Composante non connexe

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Recherche des composantes

Extraction
de routes
dans des
images
satellite

Alexandre
Havard
n° 8010

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

**Recherche des
composantes**

Filtrage

Améliorations

Ajustement des
seuils

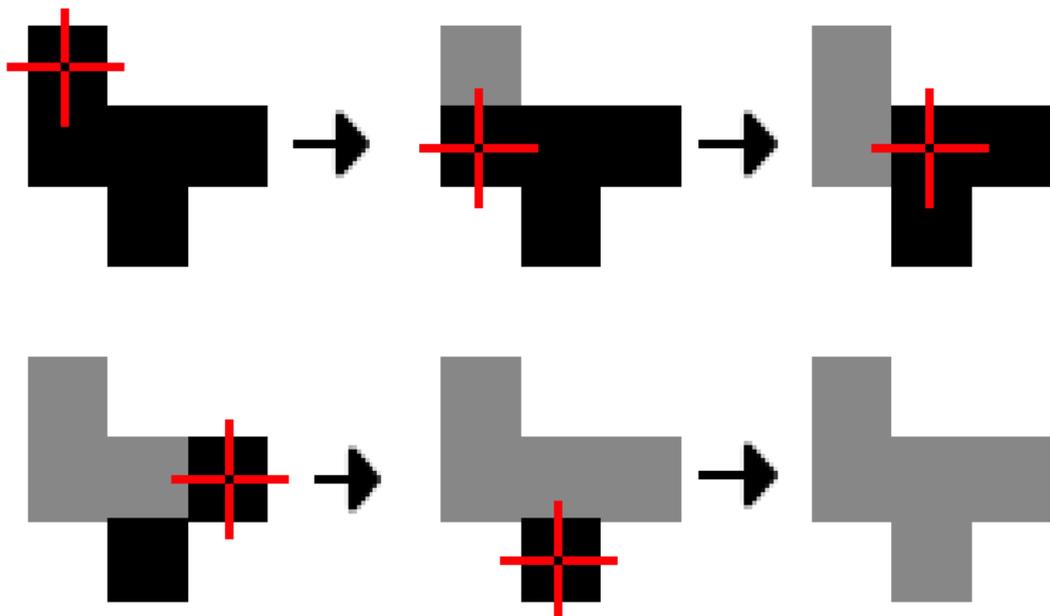
Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible



Recherche des composantes

Extraction
de routes
dans des
images
satellite

Alexandre
Havard
n° 8010

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

**Recherche des
composantes**

Filtrage

Améliorations

Ajustement des
seuils

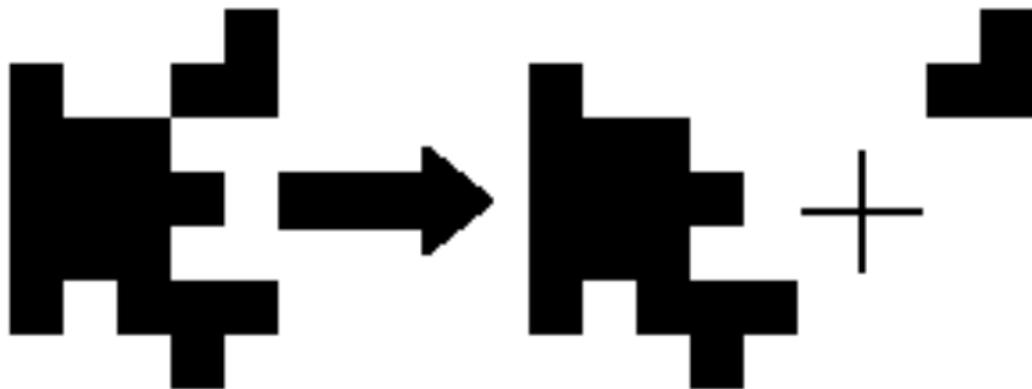
Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible



Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Filtrage

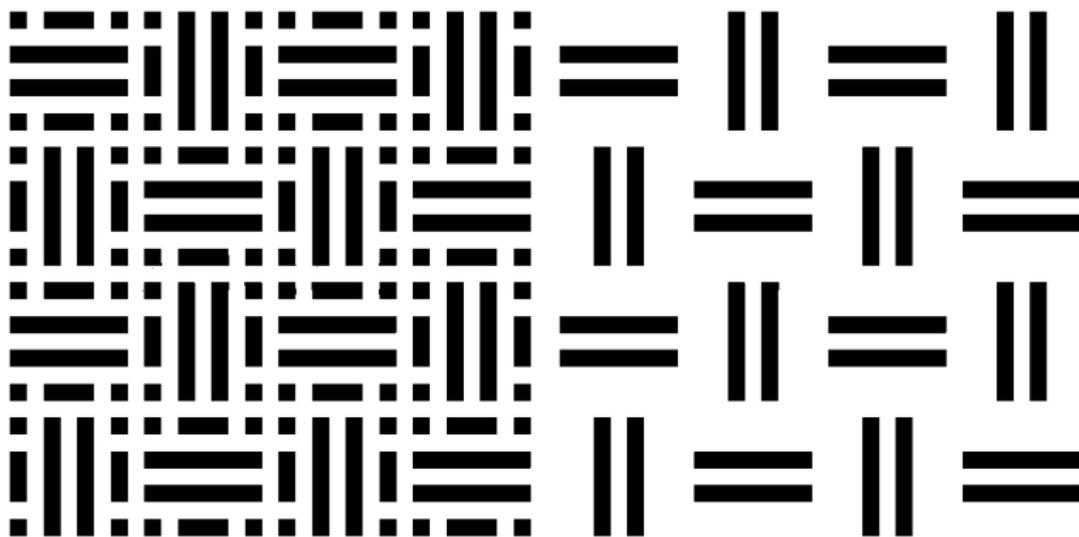


Figure – Extraction des grandes composantes connexes

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Améliorations

Extraction de routes dans des images satellite

Alexandre Havard
n° 8010

Seuillage

Initialisation

Seuil naïf

Sélection à deux seuils

Composantes connexes

Recherche des composantes

Filtrage

Améliorations

Ajustement des seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau routier

Approfondissement possible



Figure – Test de seuillage avec un seuil à 0.45, 0.55, et 0.65

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

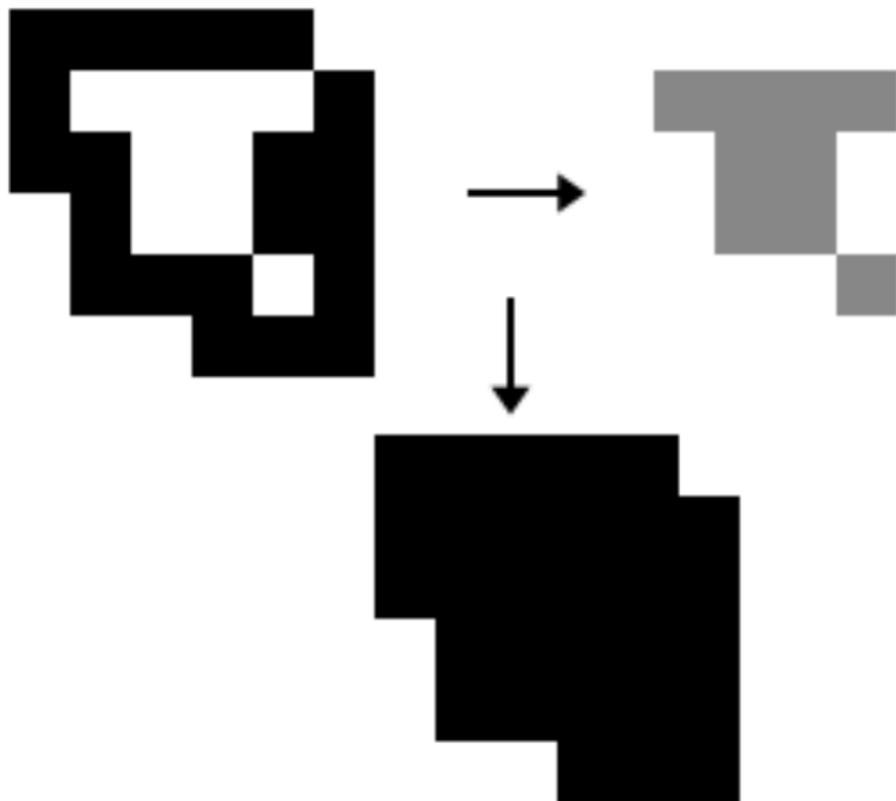
Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Remplissage



Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Extractions effectives

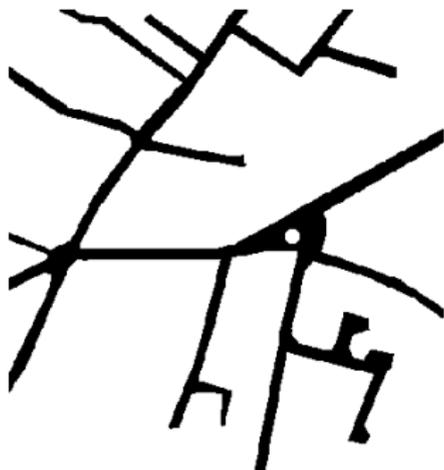


Figure – Représentation théorique des routes

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

$$C = \frac{\text{nombre de pixels trouvés sur la route}}{\text{nombre de pixels sur la route}}$$

$$F = \frac{\text{nombre de pixels trouvés hors de la route}}{\text{nombre de pixels dans l'image}}$$

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible



- Complétude : $C = 74\%$
- Faux pixels : $F = 48\%$

Figure – Seuil naïf 0.56

Extraction de routes dans des images satellite

Alexandre Havard
n° 8010

Seuillage

Initialisation

Seuil naïf

Sélection à deux seuils

Composantes connexes

Recherche des composantes

Filtrage

Améliorations

Ajustement des seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau routier

Approfondissement possible



- Complétude : $C = 61\%$
- Faux pixels : $F = 22\%$

Figure – Seuil double 0.56 et 0.67

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible



- Complétude : $C = 51\%$
- Faux pixels : $F = 4.7\%$

Figure – Filtrage des composantes
inférieures à 400 pixels

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible



Figure – Remplissage des routes

- Complétude : $C = 56\%$
- Faux pixels : $F = 5\%$

Extraction de routes dans des images satellite

Alexandre Havard
n° 8010

Seuillage

Initialisation

Seuil naïf

Sélection à deux seuils

Composantes connexes

Recherche des composantes

Filtrage

Améliorations

Ajustement des seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau routier

Approfondissement possible

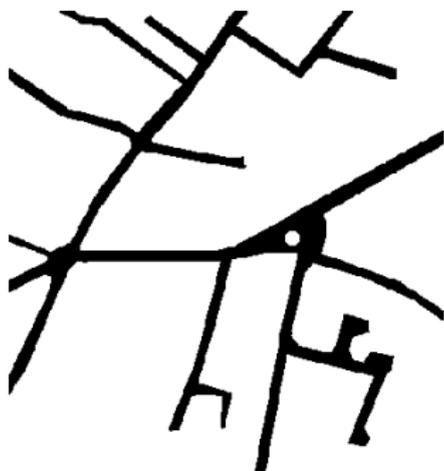


Figure – Comparaison théorique et algorithmique

Extraction
de routes
dans des
images
satellite

Alexandre
Havard
n° 8010

Photo d'un autre réseau routier

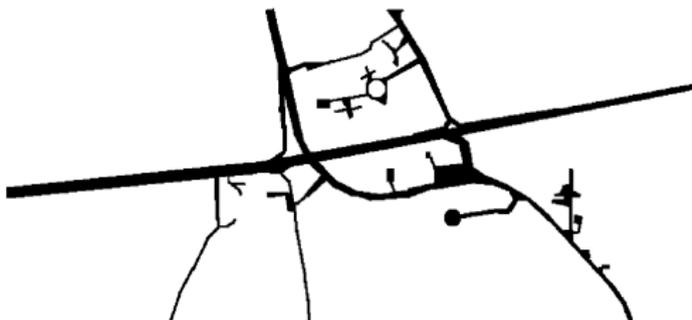


Figure – Représentation théorique des routes

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

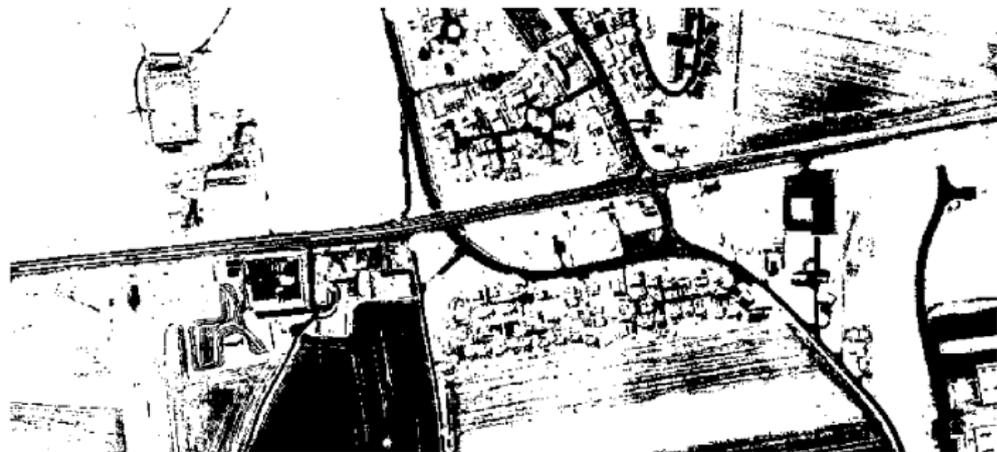
Remplissage

Extractions effectives

Noirmoutier

**Autre réseau
routier**

Approfondissement
possible



Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

**Autre réseau
routier**

Approfondissement
possible

Figure – Seuil naïf 0.61

- Complétude : $C = 74\%$
- Faux pixels : $F = 14.4\%$



Figure – Seuil double 0.61 et 0.9

- Complétude : $C = 73\%$
- Faux pixels : $F = 13,29\%$

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

Autre réseau
routier

Approfondissement
possible

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

**Autre réseau
routier**

Approfondissement
possible



Figure – Filtrage des composantes inférieures à 15000 pixels

- Complétude : $C = 69\%$
- Faux pixels : $F = 9.1\%$

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

**Autre réseau
routier**

Approfondissement
possible



Figure – Remplissage des routes

- Complétude : $C = 74\%$
- Faux pixels : $F = 10\%$

Extraction de routes dans des images satellite

Alexandre
Havard
n° 8010

Seuillage

Initialisation

Seuil naïf

Sélection à deux
seuils

Composantes connexes

Recherche des
composantes

Filtrage

Améliorations

Ajustement des
seuils

Remplissage

Extractions effectives

Noirmoutier

**Autre réseau
routier**

Approfondissement
possible

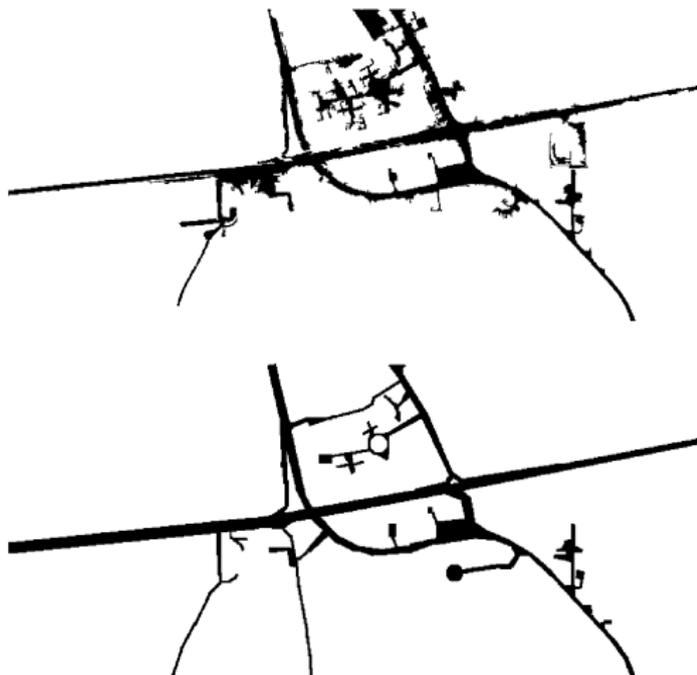


Figure – Comparaison théorique et algorithmique

Approfondissement possible

Seuillage

Initialisation
Seuil naïf
Sélection à deux
seuils

Composantes connexes

Recherche des
composantes
Filtrage

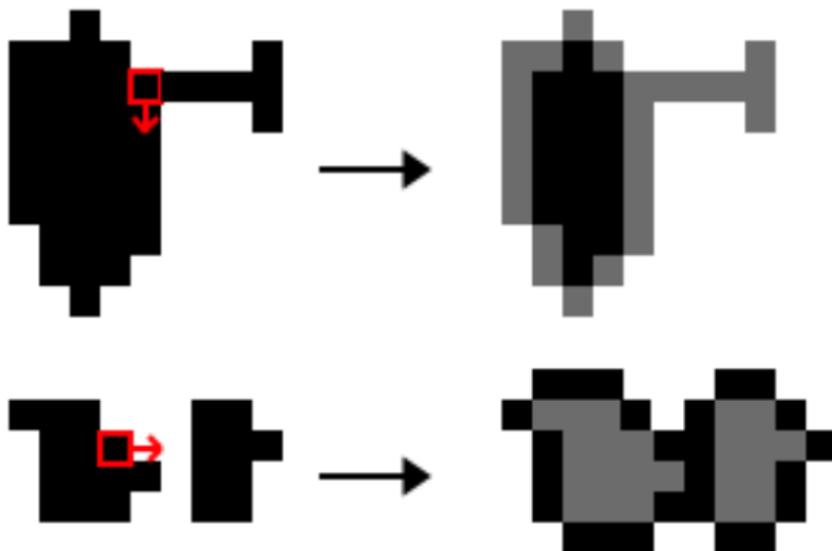
Améliorations

Ajustement des
seuils
Remplissage

Extractions effectives

Noirmoutier
Autre réseau
routier

**Approfondissement
possible**



- Érosion

- Dilatation

Code de moyennage

```
def moyenne(L):  
    S = 0  
    n = len(L)  
    for k in range (n):  
        S += L[k]  
    return S/n  
  
def transform_rgb_to_grey(image):  
    n = image.shape[0]  
    m = image.shape[1]  
    new_image = np.empty((n,m,3))  
    for i in range (n):  
        for j in range (m):  
            moy = moyenne(image[i,j])  
            for k in range (3):  
                new_image[i,j,k] = moy  
    return new_image
```

Code de seuillage naïf

```
def bicolor_seuil(image, seuil):  
    image_grey = transform_rgb_to_grey(image)  
    n = image.shape[0]  
    m = image.shape[1]  
    new_image = np.zeros((n,m,3))  
    for i in range (n):  
        for j in range (m):  
            if image_grey[i,j,1] <= seuil :  
                for k in range(3):  
                    new_image[i,j,k] = 1  
    return new_image
```

Code de seuillage double

```
def bicolor_seuil_mieux_choisi(image, seuil_min, seuil_max):  
    image_grey = transform_rgb_to_grey(image)  
    n = image.shape[0]  
    m = image.shape[1]  
    new_image = np.zeros((n,m,3))  
    for i in range (n):  
        for j in range (m):  
            if (image_grey[i,j,1] <= seuil_min  
                or image_grey[i,j,1] >= seuil_max):  
                for k in range(3):  
                    new_image[i,j,k] = 1  
    return new_image
```

Composante connexe

```
def composante_connexe_noir(image, neg=False):
    n = image.shape[0]
    m = image.shape[1]
    L_ensembles_connexes = []
    L_visite = np.full((n,m),False)
    image2= np.zeros((n,m,3))
    for z in range (n):
        for w in range (m):
            for k in range (3):
                image2[z,w,k]=1-image[z,w,k] if neg else image[z,w,k]
    for i in range (n):
        for j in range(m):
            #initialisation
            if image2[i,j,0] == 0 and L_visite[i,j] == False :
                L_noir = []
                L_attente = [(i,j)]
                L_visite[i,j] = True
```

Composante connexe

```
#sélection de la composante
while len(L_attente) != 0:
    x,y = L_attente.pop()
    L_visite[x,y] = True
    L_noir.append((x,y))
# sélection des voisins du pixel
# (en dessous et sur la droite)
    if x == (n-1) and y < (m-1) and image2[x,y+1,0] == 0 :
        L_attente.append((x,y+1))
    if y == (m-1) and x < (n-1) and image2[x+1,y,0] == 0 :
        L_attente.append((x+1,y))
    if x < (n-1) and y < (m-1) :
        if image2[x+1,y,0] == 0 :
            L_attente.append((x+1,y))
        if image2[x,y+1,0] == 0 :
            L_attente.append((x,y+1))
```

Composante connexe

```
# sélection des voisins du pixel
# (en dessus et sur la gauche)
if x == 0 and y > 0 and image2[x,y-1,0] == 0 :
    L_attente.append((x,y-1))
if y == 0 and x > 0 and image2[x-1,y,0] == 0 :
    L_attente.append((x-1,y))
if x > 0 and y > 0 :
    if image2[x-1,y,0] == 0 :
        L_attente.append((x-1,y))
    if image2[x,y-1,0] == 0 :
        L_attente.append((x,y-1))
while (len(L_attente) != 0
       and L_visite[L_attente[-1]] == True):
    L_attente.pop()
# listing des composantes
L_ensembles_connexes.append(L_noir)
return L_ensembles_connexes
```

Composantes blanches, filtrage

```
def composante_connexe_blanc(image):  
    return composante_connexe_noir(image, True)  
  
def selectionner_les_tâches_trop_grandes(ensembles_connexes,  
                                          approx):  
  
    n = len(ensembles_connexes)  
    L = []  
    for i in range (n):  
        if len(ensembles_connexes[i]) >= approx:  
            L.append(ensembles_connexes[i])  
    return L  
  
def selectionner_les_tâches_trop_petites(ensembles_connexes,  
                                          approx):  
  
    n = len(ensembles_connexes)  
    L = []  
    for i in range (n):  
        if len(ensembles_connexes[i]) <= approx:  
            L.append(ensembles_connexes[i])  
    return L
```

Filtrage

```
def retirer_le_noir_composantes_connexes(liste_tâches,image,  
                                          neg = False):  
  
    n = image.shape[0]  
    m = image.shape[1]  
    nbr = len(liste_tâches)  
    image2 = np.zeros((n,m,3))  
    for i in range (n):  
        for j in range (m):  
            for k in range (3):  
                image2[i,j,k] = image[i,j,k]  
    for p in range(nbr) :  
        for w in liste_tâches[p]:  
            x,y = w[0],w[1]  
            image2[x,y] = 0 if neg else 1  
    return image2  
  
def retirer_le_blanc_composantes_connexes(liste_tâches,image):  
    return retirer_le_noir_composantes_connexes(liste_tâches  
                                                  ,image,True)
```

Completude et Faux pixels

```
def completude_du_reseau(img,img_th):  
    n = img.shape[0]  
    m = img.shape[1]  
    nbr = len(composante_connexe_noir(img_th)[0])  
    somme = 0  
    for i in range(n):  
        for j in range(m):  
            if img_th[i,j,0] == 0 and img[i,j,0] == 0:  
                somme += 1  
    return int((somme/nbr)*100)  
  
def faux_pixels(img,img_th):  
    n = img.shape[0]  
    m = img.shape[1]  
    nbr = n*m  
    somme = 0  
    for i in range(n):  
        for j in range(m):  
            if img[i,j,0] == 0 and img_th[i,j,0] == 1 :  
                somme += 1  
    return (somme/nbr)*100
```