



ANALYSE D'IMAGE

Rapport de TD2

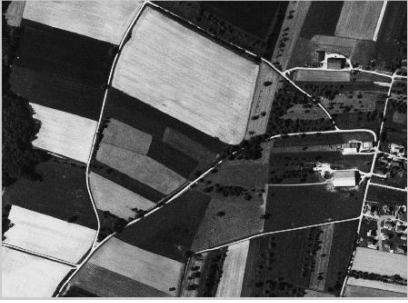

Guénon Marie et Favreau Jean-Dominique
VIM / Master SSTIM

Table des matières

Implémentation des briques élémentaires	2
Dilatation	2
Erosion.....	2
Chapeau haut de forme	3
Algorithme d'extraction du réseau routier	4
Annexes :	5
Test.m.....	5
Dilatation.m.....	5
Erosion.m	6
morphofil.m.....	6
lineique.m.....	7



Implémentation des briques élémentaires

Dilatation

Image de départ	Algorithme	Image obtenue
	<pre> for (i=1:cols) for (j=1:rows) max=tmp(i,j); for (k=i:i+xx) for (l=j:j+yy) val = tmp(k,l); if (val>max) max=val; end end end dil(i,j)=max; end end </pre>	

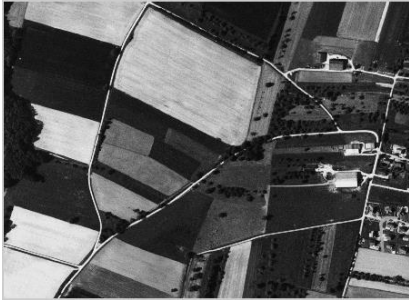

La dilatation augmente la taille des zones les plus claires, comme nous pouvons facilement le constater ici sur les routes qui ont été élargies.

Erosion

Image de départ	Algorithme	Image obtenue
	<pre> for (i=1:cols) for (j=1:rows) min=tmp(i,j); for (k=i:i+xx) for (l=j:j+yy) val = tmp(k,l); if (val<min) min=val; end end end ero(i,j)=min; end end </pre>	

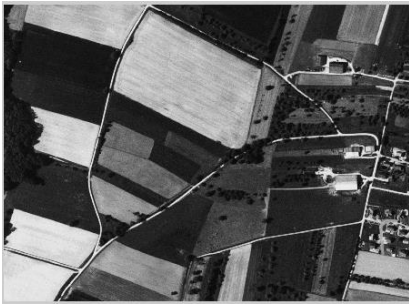

L'érosion à l'effet contraire de la dilatation et augmente la taille des zones les plus sombres, comme nous pouvons facilement le constater ici sur les routes qui ont été effacées alors que les arbres et les ombres ont été élargis.

Chapeau haut de forme

Image de départ	Algorithme	Image obtenue
	<pre>function img = close(f,x,y) img = Erosion(Dilatation(f,x,y),x,y); end function img = open(f,x,y) img = Dilatation(Erosion(f,x,y),x,y); end function img = top_hat(f,x,y) img = f-open(f,x,y) end function img = top_hat_dual(f,x,y) img = close(f,x,y) - f; end mtdt = max(top_hat(dep,x,y), top_hat_dual(dep,x,y));</pre>	

En combinant de plusieurs manières les fonctions d'érosion et de dilatation, nous obtenons une image où les routes se détachent clairement sur un fond plus sombre.

Algorithme d'extraction du réseau routier

Image de départ	Algorithme	Image obtenue
	<pre>bin=zeros(size(dep)); bin1 = morphofil(dep,1,0); bin2 = morphofil(dep,0,1); bin = bitor(hysteresis(bin1,40,140),hysteresis(bin2,40,140)); bin1 = morphofil(dep,2,0); bin2 = morphofil(dep,0,2); bin = bitor(bin ,bitor(hysteresis(bin1,50,150),hysteresis(bin2,50,150))); Afficher_extraction(dep,bin);</pre>	

Détailler pourquoi on a fait les choses dans cet ordre là

Applique morphofil sur l'image en utilisant un masque horizontal de taille 1x3 et un masque vertical de taille 3x1 ce qui nous donne deux images.

Puis nous faisons un seuillage par hysteresis sur les deux images obtenues précédemment pour mettre en évidence les lignes horizontales et respectivement verticales. Ensuite, on « or » les images, c'est-à-dire si nous avons une ligne verticale et / ou horizontale, on dit qu'il y a une ligne.

Nous recommençons cette opération avec deux masques plus grands : 1x5 et 5x1.

Enfin, nous faisons à nouveau un « or » entre les deux images obtenues avec les étapes précédentes. Et nous obtenons alors le réseau routier extrait.

Pour conclure, nous pouvons dire que en combinant plusieurs approches de top-hat et de masque, nous avons obtenu un affichage du réseau routier cohérent avec la réalité. Nous pouvons constater cependant la présence de la détection de maisons (à droite) en tant que routes, alors que certaines portions de routes s'effacent (car recouvertes par des arbres, en haut). Notre algorithme pourrait donc être amélioré, mais il semble être assez robuste sur l'image ici testée.

Annexes :

Test.m

```
function [ output_args ] = test()
%UNTITLED Summary of this function goes here
% Detailed explanation goes here
image = ima2mat('ign1');
imshow(image/255);
% Dilatation(image,1,1);
% Erosion(image,1,1);
% morphofil(image,1,1);
lineique('ign1','result');
end
```

Dilatation.m

```
function dil=Dilatation(dep,x,y)
%-----
% Entrees : dep    -> image initiale
%           x, y  -> parametres de l'element structurant
% Sortie :  dil   -> resultat de la dilatation
%-----

% ??????????
s=size(dep);
rows=s(2);
cols=s(1);
xx=2*x;
yy=2*y;
tmp = zeros([cols+x*2,rows+y*2]);
tmp(x+1:x+cols,y+1:y+rows)=dep;

for(i=1:cols)
    for(j=1:rows)
        max=tmp(i,j);
        for(k=i:i+xx)
            for(l=j:j+yy)
                val = tmp(k,l);
                if(val>max)
                    max=val;
                end
            end
            dil(i,j)=max;
        end
    end

end

% -- Affichage
figure;
photo(dil,256);
```

```
title('Dilatation');  
end
```

Erosion.m

```
function ero=Erosion(dep,x,y)  
%-----  
% Entrees : dep    -> image initiale  
%           x, y   -> parametres de l'element structurant  
% Sortie :  ero    -> resultat de l'erosion  
%-----  
  
% ??????????  
s=size(dep);  
rows=s(2);  
cols=s(1);  
xx=2*x;  
yy=2*y;  
tmp = ones([cols+x*2,rows+y*2])*255;  
tmp(x+1:x+cols,y+1:y+rows)=dep;  
  
for(i=1:cols)  
    for(j=1:rows)  
        min=tmp(i,j);  
        for(k=i:i+xx)  
            for(l=j:j+yy)  
                val = tmp(k,l);  
                if(val<min)  
                    min=val;  
                end  
            end  
        end  
        ero(i,j)=min;  
    end  
end  
% -- Affichage  
figure;  
photo(ero,256);  
title('Erosion');  
end
```

morphofil.m

```
function mtdt=morphofil(dep,x,y)  
%-----  
% function mtdt=morphofil(dep,x,y)  
%  
% Max entre le Top hat de l'image name1 et son dual  
%  
% Entrees : dep    -> image de depart  
%           x,y     -> pour definir l'element structurant plan (rectangle de  
%                   dimension (2x+1)x(2y+1) centre en 0)  
%  
%
```

```
% Sortie : mtdt -> max entre top hat et son dual
%-----

% ???????
function img = close(f,x,y)
    img = Erosion(Dilatation(f,x,y),x,y);
end
function img = open(f,x,y)
    img = Dilatation(Erosion(f,x,y),x,y);
end
function img = top_hat(f,x,y)
    img = f-open(f,x,y)
end
function img = top_hat_dual(f,x,y)
    img = close(f,x,y) - f;
end
mtdt = max(top_hat(dep,x,y),top_hat_dual(dep,x,y));
% -- Affichage
figure;
photo(mtdt,256);
title('Max entre top hat et son dual');
end
```

lineique.m

```
function bin=lineique(name1,name2)
close all;
%-----
% function image=lineique(name1,name2,bas,haut)
%
% Extraction de reseau routier par morphologie mathematique
%
% Entrees : name1 -> nom de l'image (.ima) a traiter (si l'image a traiter
%               s'appelle ign1.ima l'appel se fait avec name1='ign1')
%               name2 -> nom pour enregistrer l'image en sortie
%
%               %%% Ajoutez les parametres que vous juges necessaires !!!!!
%
% Sortie : bin -> image binarisee du reseau routier
%-----

%-- Lecture de l'image de depart
disp(name1);
%dep = ima2mat('ign1');
dep = ima2mat(name1);
% ?????????
%tmp=dep;
bin=zeros(size(dep));
%figure;imshow(tmp/255);
bin1 = morphofil(dep,1,0);
bin2 = morphofil(dep,0,1);
bin = bitor(hysteresis(bin1,40,140),hysteresis(bin2,40,140));
bin1 = morphofil(dep,2,0);
bin2 = morphofil(dep,0,2);
bin = bitor(bin ,bitor(hysteresis(bin1,50,150),hysteresis(bin2,50,150)));
%bin1 = morphofil(dep,3,0);
%bin2 = morphofil(dep,0,3);
%bin = bitor(bin ,bitor(hysteresis(bin1,80,150),hysteresis(bin2,80,150)));
```



```
%tmp=bitand(tmp,bin);  
  
%bin  
  
%bin =;%filtr vert et filtre horizontal  
  
% -- Affichage du resultat  
%figure;imshow(tmp);  
%figure;  
Afficher_extraction(dep,bin);  
  
% -- Ecriture du resultat  
mat2ima(bin,name2);
```