

Code de Freeman et morphologie mathématique

Sommaire

4.1 Morphologie mathématique binaire	94
4.1.1 Addition de Minkowski et dilatation	94
4.1.2 Erosion, ouverture et fermeture	95
4.2 Morphologie mathématique et code de Freeman	97
4.2.1 Première approche de la dilatation	97
4.2.2 Nouvelle approche de la dilatation	99
4.2.3 Approches de l'érosion	105
4.3 Conclusion	107

La théorie de la morphologie mathématique binaire est basée sur les opérations ensemblistes de Minkowski, et a été introduite dans les années 1960 par Matheron [Mat] et Serra [Ser82] et [Ser88].

Nous présentons ici un résumé de cet aspect du traitement d'image qui peut être complété par l'ouvrage de Coster et Chermant [Cos].

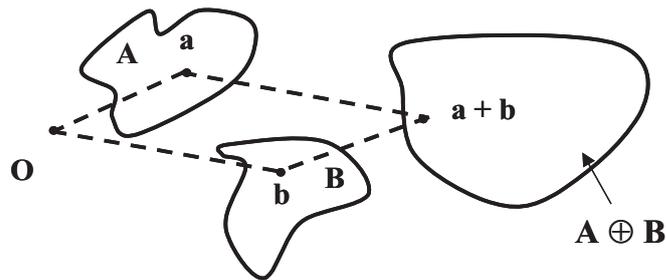


FIGURE 4.1 – Addition de Minkowski.

4.1 Morphologie mathématique binaire

4.1.1 Addition de Minkowski et dilatation

La base mathématique de la morphologie mathématique se trouve dans l'addition de Minkowski (1897) :

soient :

- deux parties quelconques A et B de \mathbb{R}^2
- l'origine O de \mathbb{R}^2

Au couple (A, B) et à O , on associe $A \oplus B = \bigcup(a + b)$, $a \in A, b \in B$, où $a + b$ désigne l'extrémité du vecteur $\overrightarrow{Oa} + \overrightarrow{Ob}$. Une illustration de ce principe est présentée figure 4.1.

Cette addition n'utilise donc que l'addition vectorielle et permet d'ajouter différents ensembles entre eux, sans qu'ils aient a priori de points communs, que ce soit au niveau de la forme ou de la taille.

Prenons maintenant O' comme origine. Pour tout $a \in A$ et $b \in B$, on a alors :

$$\overrightarrow{O'a} + \overrightarrow{O'b} = \overrightarrow{O'O} + \overrightarrow{Oa} + \overrightarrow{O'O} + \overrightarrow{Ob} = \overrightarrow{Oa} + \overrightarrow{Ob} + 2\overrightarrow{O'O}.$$

Ainsi, le résultat $A \oplus B$ est indépendant du choix de l'origine, à une translation près : $2\overrightarrow{O'O}$. De plus, si la partie B est réduite à un seul point b , on obtient A_b le translaté de A par b :

$$A \oplus B = \bigcup_{a \in A}(a + b) = (\bigcup_{a \in A} a) + b = A + b = A_b.$$

La morphologie mathématique binaire n'utilise qu'un cas particulier de cette addition de Minkowski en faisant jouer un rôle différent à A et B :

- A est l'objet à étudier.
- B est appelé l'élément structurant choisi par l'utilisateur.

Dans le cadre de notre étude d'objet grâce au code de Freeman, A est un objet blanc sur fond noir, issu de la binarisation et étudié avec le codage de Freeman ; B est ici un disque centré en O . Nous ne verrons plus loin que le cas des 2 disques élémentaires principaux, à savoir le voisinage V_4 et le voisinage V_8 dans le cadre discret.

En considérant ce cas particulier, $A \oplus B$ est appelée la dilatation de A par B et est illustrée sur la figure 4.2 page suivante. Sur cette figure, a_1 , a_2 et a_3 jouent chacun un rôle différent :

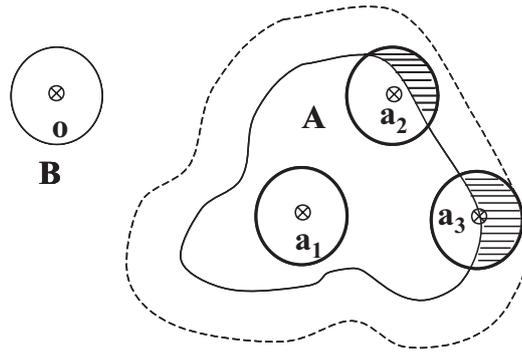


FIGURE 4.2 – Dilatation de A par B.

- $a_1 \in A$
- a_2 ajoute à $A \oplus B$ une petite partie de A^C
- a_3 est situé sur la frontière de A et apporte à $A \oplus B$ une portion de disque de rayon r_B .

Ceci s'explique simplement en écrivant :

$$A \oplus B = \bigcup_{a \in A} (\bigcup_{b \in B} (a + b)) = \bigcup_{a \in A} (B_a)$$

c'est-à-dire que le dilaté de A par B n'est autre que l'union des translatés de B par tous les $a \in A$.

De plus, les points a de type a_3 (voir figure 4.2) appartenant à la frontière de A , notée $Fr(A)$, jouent le rôle le plus important dans la dilatation de A par B . On peut donc écrire :

$$A \oplus B = A \cup (\bigcup_{a \in Fr(A)} (B_a)).$$

En imaginant maintenant l'élément structurant B roulant sur la frontière de l'objet à étudier A , la trajectoire du centre de B donne la frontière du dilaté de A .

On a donc : $Fr(A \oplus B) = \{x \in \mathbb{R}^2, d(x, A) = r_B\}$ et $A \oplus B = \{x \in \mathbb{R}^2, d(x, A) \leq r_B\}$.

Ainsi, des dilatations successives propageront à partir de A une distance par pas de r_B .

La dilatation :

- est commutative : $A \oplus B = B \oplus A$,
- ne conserve pas la connexité : le dilaté de deux objets proches "géographiquement" peut ne former qu'un seul objet,
- est invariante par translation : $A_x \oplus B = (A \oplus B)_x$,
- est compatible avec la multiplication par un réel λ , symbolisant une homothétie de rapport λ : $\lambda(A \oplus B) = \lambda A \oplus \lambda B$,
- est croissante : $A \subseteq A' \Rightarrow A \oplus B \subseteq A' \oplus B$,
- est extensive : $A \subseteq (A \oplus B)$.

4.1.2 Erosion, ouverture et fermeture

L'érodé de A par B , noté $A \ominus B$ et représenté figure 4.3 page suivante, est défini par :

$$A \ominus B = (A^C \oplus B)^C$$

De même que pour le dilaté, l'érodé peut être obtenu simplement en considérant la trajectoire du centre de l'élément structurant B roulant à l'intérieur de l'objet étudié A .

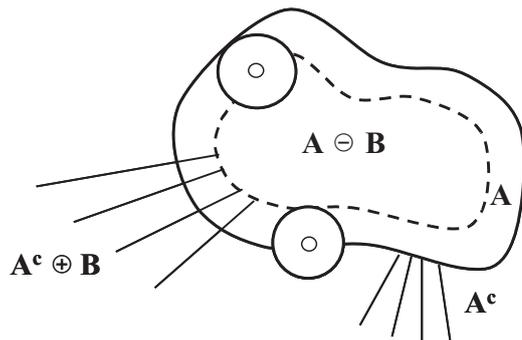


FIGURE 4.3 – Erosion de A par B.

L'érosion :

- est commutative : $A \ominus B = B \ominus A$,
- ne préserve pas la connexité,
- est invariante par translation : $A_x \ominus B = (A \ominus B)_x$,
- est compatible avec la multiplication par un réel λ , symbolisant une homothétie de rapport λ : $\lambda(A \ominus B) = \lambda A \ominus \lambda B$,
- est croissante : $A \subseteq A' \Rightarrow A \ominus B \subseteq A' \ominus B$,
- est anti-extensive : $(A \ominus B) \subseteq A$.

L'érosion et la dilatation peuvent être composées de deux façons différentes :

- L'ouverture de A par B est le dilaté de l'érodé de A. On la note $O(A, B) = (A \ominus B) \oplus B$.
L'ouverture :
 - ne préserve pas la connexité,
 - est une opération croissante : $A \subseteq A' \Rightarrow O(A, B) \subseteq O(A', B)$,
 - est anti-extensive : $O(A, B) \subseteq A$,
 - est idempotente : $O(O(A, B), B) = O(A, B)$.
- La fermeture de A par B est l'érodé du dilaté de A. On la note $F(A, B) = (A \oplus B) \ominus B$.
La fermeture :
 - ne préserve pas la connexité,
 - est une opération croissante : $A \subseteq A' \Rightarrow F(A, B) \subseteq F(A', B)$,
 - est extensive : $A \subseteq F(A, B)$,
 - est idempotente : $F(F(A, B), B) = F(A, B)$.

La particularité de ces deux opérateurs se trouve dans les différences : $A \setminus [(A \ominus B) \oplus B]$ et $[(A \oplus B) \ominus B] \setminus A$ qui mettent en évidence respectivement les zones sortantes et rentrantes de A, schématisées sur la figure 4.4 page ci-contre. Ces résultats sont dépendants de la taille et de la forme de l'élément structurant.

Pour un élément structurant donné, les opérateurs de morphologie mathématique permettent de compter le nombre de zones rentrantes et sortantes d'un objet, et ainsi déterminer par exemple le nombre d'objets présents dans un amas ou de trier des objets par le nombre de zones sortantes qu'ils possèdent.

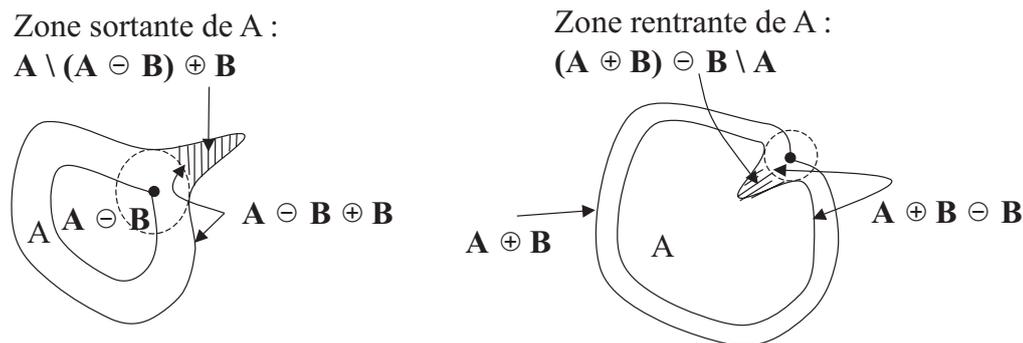


FIGURE 4.4 – Zones sortantes et rentrantes de A.



FIGURE 4.5 – Codant 6 et sa saucisse englobante.

4.2 Morphologie mathématique et code de Freeman

4.2.1 Première approche de la dilatation

Le but est ici d'obtenir, à partir du code de Freeman d'un objet, les codes de Freeman de son érodé et de son dilaté au sens de la morphologie mathématique. Par la suite, nous pourrons utiliser toutes les méthodes de traitement d'objets au sens morphomathématique, directement appliquées sur le code de Freeman.

Comme pour l'ensemble de ce manuscrit, nous étudions ici les objets sans trou car le codage de Freeman simple ne détecte pas ces derniers.

L'idée de départ de David Odin [Odi] pour rechercher le dilaté par un voisinage V_8 est de coder la "saucisse" englobante de chaque codant. C'est-à-dire qu'à un codant, on associe la chaîne de codants faisant le tour du codant de départ. Ainsi, l'exemple de la figure 4.5 montre le codant 6 et son dilaté : 6660022244. Les cas des codants pairs et impairs étant très différents, la saucisse englobante du codant 1 est 5660012244.

Il n'est cependant pas possible d'associer à chaque codant cette saucisse englobante car il faut tenir compte du code précédent dans le cas général et des codes précédents dans quelques cas particuliers.

Pour obtenir cette "saucisse", on se propose de l'initialiser au premier codant comme dans l'exemple ci-dessus. Par la suite, on propage ce point de départ à la suite des codants de l'objet. Pour finir, on réajustera la fin et le début du code de la "saucisse" englobante suivant des règles bien particulières.

Les cas dépendent des huit directions du code de Freeman, mais on peut réduire ce nombre de cas par simple rotation d'un angle multiple de 45° . (Rappelons ici qu'une rotation du code

de Freeman d'un angle de $k \times 45^\circ$ correspond à ajouter $k[8]$ à chaque codant du code de Freeman, voir § 3.1.3 page 75.)

On se base alors sur les codants 0 et 1 qui ont des voisinages V_8 associés très différents. On analyse ensuite le codant suivant et le codant précédent. (Voir le tableau de la figure 4.6)

Codant	précédent	début[-]	début[+]	fin[-]	fin[+]
0-6		-	022	-3	60
0-7		-	224	-2	600
0-0		-1	24	-1	02
0-1		-2	446	-	022
0-2	6	-3	46660222	-3	002224
0-2	$\neq 6$	-3	46	-	224
0-3	7	-5	24244	-	-
0-3	$\neq 7$	-4	46	-	2424
0-4	-	-	60002	-	24446
0-5	3	-5	00202	-	-
0-5	$\neq 3$	-4	00202	-	6
1-7		-	2242	-4	00
1-0		-	24	-2	02
1-1		-1	446	-1	022
1-2		-2	46	-	24
1-3		-4	66	-	4244
1-4		-	-	-	-
1-5		-3	00224	-3	24466
1-6		-	-	-	-

FIGURE 4.6 – Obtention de la saucisse de dilatation pour les codants 0 et 1. [-] : nombre de codants que l'on enlève au code de Freeman courant ; [+] : codants que l'on ajoute ensuite au code de Freeman courant.

Après le déroulement de l'algorithme, on obtient le code de Freeman de la "saucisse" du contour. Celle-ci est bien fermée, mais il y a superposition de la "saucisse" sur elle-même au niveau du premier et du dernier codant du code étudié.

Cette méthode possède donc ses limites, notamment dans la déduction de l'érodé et du dilaté. Il semble possible d'adopter une autre approche qui consisterait à ne plus utiliser une "saucisse" dès le premier codant, mais à suivre le contour (voir § 4.2.2 page ci-contre).

De plus, nous verrons les cas de dilatation et d'érosion en voisinage V_4 et en voisinage V_8 . Ici, David Odin n'a travaillé qu'en voisinage V_8 .

Nous avons vu que cette méthode présente des limites, c'est pourquoi nous nous attacherons à trouver une autre méthode permettant d'obtenir le dilaté et l'érodé d'un objet en travaillant uniquement sur le code de Freeman de son contour. Nous étudierons également la notion d'érodé ultime à partir de cette méthode.

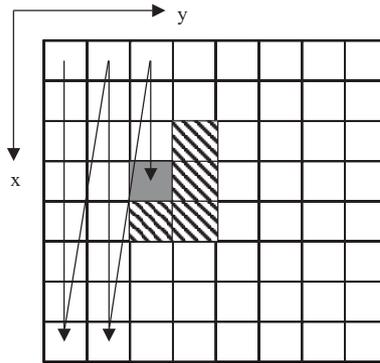


FIGURE 4.7 – Pixel de départ et premiers codants possibles.

4.2.2 Nouvelle approche de la dilatation

4.2.2.1 Dilatation par un voisinage V_4

Le but ici est d'aborder la notion de morphologie mathématique à partir du code de Freeman du contour d'un objet, selon une méthode différente. On ne considère donc pas l'objet complet mais seulement la couronne représentant le code de Freeman de la frontière interne de l'objet. Le dilaté par un voisinage V_4 de l'objet correspond donc à la couronne entourant l'objet de départ. Connaissant le code de Freeman de la première couronne, il est possible d'obtenir le code de la seconde, en travaillant uniquement sur la chaîne de codants.

Cas du premier point :

Rappelons que le premier point du contour de l'objet est obtenu par balayage lexicographique selon la figure 4.7 : le pixel foncé est le premier pixel allumé trouvé, et les pixels hachurés correspondent aux pixels voisins possibles. En effet, dans ces conditions de balayage représenté par les flèches, le premier codant du code de Freeman du contour d'un objet ne peut être que 6, 7, 0 ou 1.

Les coordonnées du premier point du code de Freeman du contour de l'objet dont on désire le code du dilaté sont (x, y) . Prenons comme premier point du dilaté le point de coordonnées $(x, y - 1)$. Ce point nous paraît judicieux car avec les mêmes conventions de balayage, il correspond au point de départ réel du dilaté de l'objet par un voisinage V_4 avec une méthode classique de dilatation. Il s'agit donc du pixel situé immédiatement à gauche du pixel foncé sur la figure 4.7.

Cas du premier codant :

Sur la figure 4.8 page suivante, le premier codant du code à dilater est représenté par des pixels, celui de départ étant hachuré et celui d'arrivée étant vide. Le codage du dilaté est représenté par des flèches.

Il faut s'arrêter ici car les codants pouvant suivre le codant 6 (en trait plein sur la figure 4.9 page suivante) sont 5, 6, 7, 0, 1 et 2 (en pointillé). Le code du dilaté (en gras) doit s'arrêter avant les pixels pouvant être atteints par le dilaté du codant suivant. Les autres cas sont identiques et obtenus par rotation de ce cas donné en exemple.

Illustration Codant de départ Codant du dilaté

	6	6
	7	7
	0	70
	1	71

FIGURE 4.8 – Dilatation par un voisinage V_4 des premiers codants possibles.

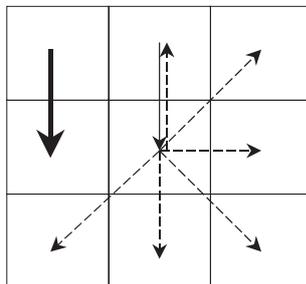


FIGURE 4.9 – Arrêt du codage de la dilatation par un voisinage V_4 pour le codant 6.

Cas général :

Le code du dilaté dépend bien entendu du codant actuel, mais également du codant précédent. Pour chaque cas, il suffit d'ajouter au code du dilaté les codants selon le tableau de la figure 4.10.

Ancien codant	Nouveau codant							
	0	1	2	3	4	5	6	7
0	0	1	12	13	134	X	X	-
1	0	1	12	13	134	135	X	-
2	X	-	2	3	34	35	356	X
3	X	-	2	3	34	35	356	357
4	570	X	X	-	4	5	56	57
5	570	571	X	-	4	5	56	57
6	70	71	712	X	X	-	6	7
7	70	71	712	713	X	-	6	7

FIGURE 4.10 – Cas général de la dilatation par un voisinage V_4 . X : cas impossible; - : le code reste inchangé.

Cas du dernier codant :

Après avoir appliqué le cas général au dernier codant, il faut également lui appliquer le correctif représenté sur la figure 4.11 page suivante, afin de revenir au point de départ de l'objet dilaté choisi plus haut. Le dernier codant à dilater est représenté par des pixels, le départ étant un pixel vide et l'arrivée un pixel hachuré. Le dilaté est représenté par des flèches.

Ainsi, en déroulant le code du contour de l'objet, et en respectant les tableaux ci-dessus, on obtiendra le code du dilaté par un voisinage V_4 de la forme considérée.

4.2.2.2 Dilatation par un voisinage V_8

Le principe appliqué ici est exactement le même que celui de la dilatation par un voisinage V_4 . Il suffit simplement de se référer aux tableaux suivants : 4.12 page suivante pour le cas du premier codant, 4.13 page suivante pour le cas général et 4.14 page 103 pour les codants à ajouter après le dernier cas général.

Il est amusant dans le tableau de la figure 4.13 page suivante de remarquer que pour passer de la dilatation par un voisinage V_4 à la dilatation par un voisinage V_8 , on ne fait que remplacer un code à huit directions par le code à quatre directions le plus proche lui correspondant. En effet, par exemple, le codant à huit directions 13 (correspondant à la dilatation du codant 3 précédé du codant 0) aurait été codé par 0224 si l'on n'avait utilisé que les directions paires.

Illustration	Dernier codant	Derniers codants du dilaté
	2	2 3 5
	3	3 5
	4	4 5
	5	5

FIGURE 4.11 – Derniers codants à ajouter pour boucler la dilatation par un voisinage V_4 .

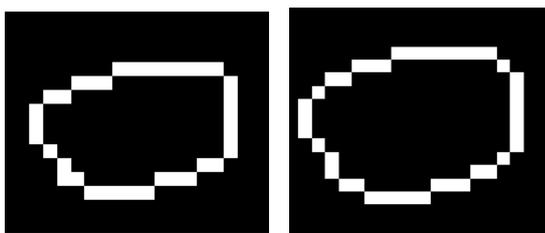
Codant de départ	Codant(s) du dilaté
6	6 6
7	6 6 0
0	6 6 0 0
1	6 6 0 0 2

FIGURE 4.12 – Dilatation par un voisinage V_8 des premiers codants possibles.

Ancien codant	Nouveau codant							
	0	1	2	3	4	5	6	7
0	0	02	022	0224	02244	X	X	-
1	0	02	022	0224	02244	022446	X	-
2	X	-	2	24	244	2446	24466	X
3	X	-	2	24	244	2446	24466	244660
4	46600	X	X	-	4	46	466	4660
5	46600	466002	X	-	4	46	466	4660
6	600	6002	60022	X	X	-	6	60
7	600	6002	60022	600224	X	-	6	60

FIGURE 4.13 – Cas général de la dilatation par un voisinage V_8 . X : cas impossible ; - : le code reste inchangé.

Codant de départ	Codant(s) du dilaté
2	244
3	244
4	4
5	4

FIGURE 4.14 – Derniers codants à ajouter pour boucler la dilatation par un voisinage V_8 .FIGURE 4.15 – Représentation de la frontière d'un objet (à gauche) et de la frontière de son dilaté par un voisinage V_4 (à droite).

4.2.2.3 Problèmes liés à cette méthode

Premier problème :

Regardons d'un peu plus près ce que nous obtenons : dans le cas d'un objet convexe, la dilatation, qu'elle soit faite avec un voisinage V_4 ou un voisinage V_8 , ne pose aucun problème. Un exemple de résultat correct d'un objet dilaté par un voisinage V_4 est présenté sur la figure 4.15.

Le contour de l'objet de gauche a pour code 667760700001001012222234444444544545, et le dilaté de ce contour est codé par : 6677670700001001011222223344444445445455.

En revanche, pour certains objets un peu plus complexes, tels que l'étoile représentée figure 4.16, il y a quelques imperfections à corriger. En effet, on s'aperçoit qu'en appliquant la méthode vue précédemment, on obtient le résultat présenté sur cette même figure.

FIGURE 4.16 – Représentation de la frontière d'une étoile (à gauche), de la frontière de son dilaté par un voisinage V_4 suivant notre méthode (au centre) et de la frontière du dilaté souhaité (à droite).

Ceci provient du fait que la méthode proposée ne prend pas en compte les concavités. Une solution pour y remédier est de corriger le code de l'objet dilaté afin de supprimer ce



FIGURE 4.17 – Les séquences 6 4 ou 7 4 4 doivent être remplacées par le codant 5.

Séquences impossibles	Codant correspondant
05	6
06	7
16	0
27	0
20	1
30	2
41	2
42	3
52	4
63	4
64	5
74	6

FIGURE 4.18 – Séquences impossibles et codants correspondants.

défaut. Pour cela, il suffit de parcourir le code en recherchant les suites de codants impossibles compte tenu du sens de rotation choisi au départ. Ces séquences correspondent à celles déjà évoquées figure 1.23 page 18. Nous allons ici les compléter dans le tableau de la figure 4.18 et par l'exemple 4.17.

Il existe alors plusieurs possibilités pour corriger le code, mais la plus intéressante consiste en l'algorithme 18.

Algorithme 18 Correction de code après dilatation

```

1 ancien code = code à corriger
2 nouveau code = [] (code vide)
3 tant que nouveau code  $\neq$  ancien code faire
4   nouveau code = ancien code
5   Recherche des séquences impossibles dans ancien code
6   si séquence impossible trouvée alors
7     remplacement par le code correspondant dans le nouveau code
8   fin si
9 fin tant que
10 retourner nouveau code

```

Second problème :

Il existe un second problème lié à cette méthode de dilatation. Nous l'illustrons figure 4.19 page suivante.



FIGURE 4.19 – Représentation de la frontière d’un objet (à gauche), de la frontière de son dilaté par un voisinage V_4 suivant notre méthode (au centre) et de la frontière du dilaté souhaité (à droite).

Le dilaté de l’objet étudié présente la particularité de passer deux fois au même endroit : la dilatation d’une partie de l’objet se superpose avec la dilatation d’une autre partie de l’objet. La solution pour corriger ce problème consiste à suivre la chaîne de codants et à vérifier si elle passe plusieurs fois au même endroit. Si c’est le cas, il faut raccorder correctement les 2 séquences de la chaîne et considérer la chaîne du dilaté en 2 parties. Nous obtenons donc un objet avec un trou, nécessitant une structure de code particulière que nous n’étudions pas ici.

Il est également possible d’étudier la dilatation conditionnelle avec cette méthode : pour cela, il suffit de considérer la dilatation comme nous venons de la voir et de réaliser l’intersection avec la région d’intérêt grâce à la méthode d’intersection vue au § 3.2.2.1 page 81.

4.2.3 Approches de l’érosion

4.2.3.1 Dilatation du complémentaire

Nous avons vu que l’érosion d’un objet est définie comme le complémentaire du dilaté du complémentaire de cet objet (§ 4.1.2 page 95). De la même façon, nous obtenons le code de Freeman de l’érodé d’un objet selon l’algorithme 19.

Algorithme 19 Calcul du code de l’érodé d’une forme

- 1 obtention du complémentaire de A ,
 - 2 dilatation de ce complémentaire par la méthode vue au § 4.2.2 page 99,
 - 3 calcul du complémentaire de ce dilaté.
-

L’obtention du complémentaire d’un objet grâce au code de Freeman s’obtient simplement en prenant la chaîne inverse (voir § 2.1.1 page 26). En effet, d’après notre convention, lorsque l’on suit la chaîne de Freeman du contour d’un objet, on laisse celui-ci sur la gauche. En tournant dans le sens inverse, c’est-à-dire en prenant la chaîne inverse, on a bien l’objet de départ à droite et le complémentaire à gauche.

4.2.3.2 Intersection de translats

Le dilaté d’un objet A par un voisinage V_4 peut se faire en prenant l’union des translats de A dans chacune des 4 directions données par ce voisinage. De même, l’érodé s’obtient en prenant l’intersection de ces translats.

La translation d’un objet décrit par le code de Freeman de son contour (§ 3.1.1 page 70) et les algorithmes d’intersection et d’union (§ 3.2.2.1 page 81 et § 3.2.2.2 page 86) étant

connus, les codes de Freeman de l'érodé et du dilaté peuvent être obtenus de cette façon grâce au code de Freeman du contour de l'objet initial.

4.2.3.3 Erodé ultime

L'algorithme 20 donne l'obtention de l'érodé ultime d'un objet A par un élément structurant B .

Algorithme 20 Calcul du code de l'érodé ultime d'une forme

```

1 Acquisition de  $A$ 
2  $C = A$ 
3  $D = \{\emptyset\}$ 
4 tant que aire( $C$ ) > 0 faire
5    $A = R(C, A)$ 
6    $A = C \setminus A$ 
7    $D = D \cup A$ 
8    $C = C \ominus B$ 
9    $A = C$ 
10 fin tant que
11 retourner  $D$ 

```

C et D sont des mémoires temporaires nécessaires à l'exécution de cet algorithme. L'érodé ultime se trouve au final dans D . Il faut noter ici que $R(C, A)$ désigne la reconstruction par marqueur de C par A . Cette reconstruction par marqueur nécessite les algorithmes de dilatation et d'intersection tels que nous les avons étudiés.

L'algorithme 21 présente la reconstruction par marqueur.

Algorithme 21 Reconstruction par marqueur

```

1 Acquisition de  $A$  (image binaire à reconstruire)
2 Acquisition de  $M$  (masque de reconstruction)
3  $M = M \cap A$  (on conserve uniquement les marqueurs utiles)
4  $r = \text{aire}(M)$ 
5  $s = 0$ 
6 tant que  $r \neq s$  faire
7    $s = r$ 
8    $M = A \cap (M \oplus B)$ 
9    $r = \text{aire}(M)$ 
10 fin tant que
11 retourner  $M$ 

```

Lors de l'exécution de ces algorithmes, il faut faire particulièrement attention à la structure de données utilisée pour stocker le code de Freeman des objets et bien considérer les points isolés introduits par exemple par l'algorithme de calcul de l'intersection de deux objets.

4.3 Conclusion

Le code de Freeman permet donc de réaliser les opérations de base de la morphologie mathématique binaire. Les méthodes étudiées ici sont assez rapides pour des objets convexes et s'avèrent nettement plus lentes pour des objets complexes nécessitant une correction du code obtenu.