RDF TP2 : codage d'un contour

binôme : Benjamin Ruytoor et Aurore Allart
date : 26 janvier 2013

Introduction :

Notre problème pour reconnaître des formes dans une image est qu'on ne puisse pas pour l'instant définir un contour distincts de ce forme vis à vis des autres.

Dans ce TP, nous allons justement aborder 2 méthodes pour définir un contour. Soit en codant la signature de façon plus compacte grâce aux Descripteurs de Fourier, soit en éliminant des points pour les rapprocher d'un segment grâce à l'algorithme de la corde.

1. Descripteurs de Fourier

Dans cette première partie, nous allons aborder les descripteurs de Fourier. Les descripteurs de Fourier sont les coefficients de la transformée de Fourier discrète. La transformée de Fourier permet d'associer à chaque contour un spectre en fréquences.

Voici les représentations d'un contour d'un cercle en prenant en compte soit tous les points du contour (rouge), soit un point sur 4 (bleu) et soit avec un point sur 8 (vert) :

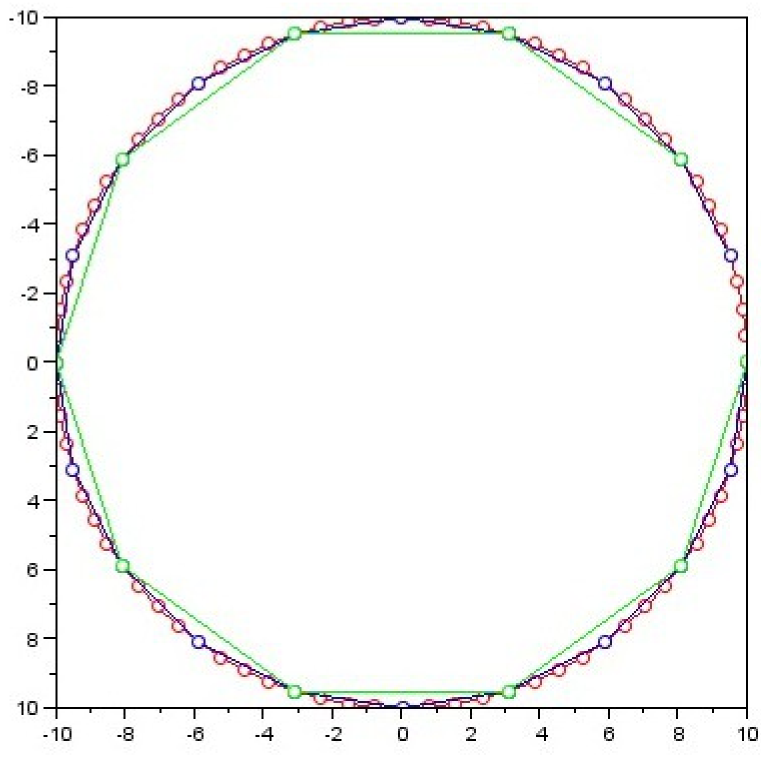

Figure 1 : représentations d'un contour d'un cercle.

Comme on peut le constater, même en prenant un point sur 8, on peut facilement dire que c'est le contour d'un cercle. Nous verrons par la suite, qu'il en va autrement avec un carré. La troncature du nombre de descripteur de Fourier permet de représenter des formes plus lisses qui restent fermées.

La fonction rdfDescFourier élimine parfois un point de la liste, quand celle-ci est de taille impair car chaque coefficient jouent 2 à 2 des rôles symétriques mais opposés sauf pour Z0 et Z1, qui sont le barycentre et le facteur d'échelle.

Le barycentre Z0 est à l'indice 1 dans la matrice résultante de la fonction rdfDescFourier. La fonction rdfValeurDescFourier permet de retourner la valeur dans la matrice à l'indice demandée. Par exemple, si on demande de retourner dans les descripteurs de Fourier la valeur de l'indice 0 (lien avec l'indice de Z), il retournera le barycentre. De plus, si on impose Z0 = 0, la description devient invariant par la translation.

Si on modifie la valeur de Z0 dans la matrice de descripteurs avant l'inversion, cela aura pour effet de translater la forme. On peut constater dans la figure 2, le décalage entre le carré noir (le carré initial), et celui de l'inversion (le rouge). On voit nettement l'influence du barycentre. En voici un exemple :

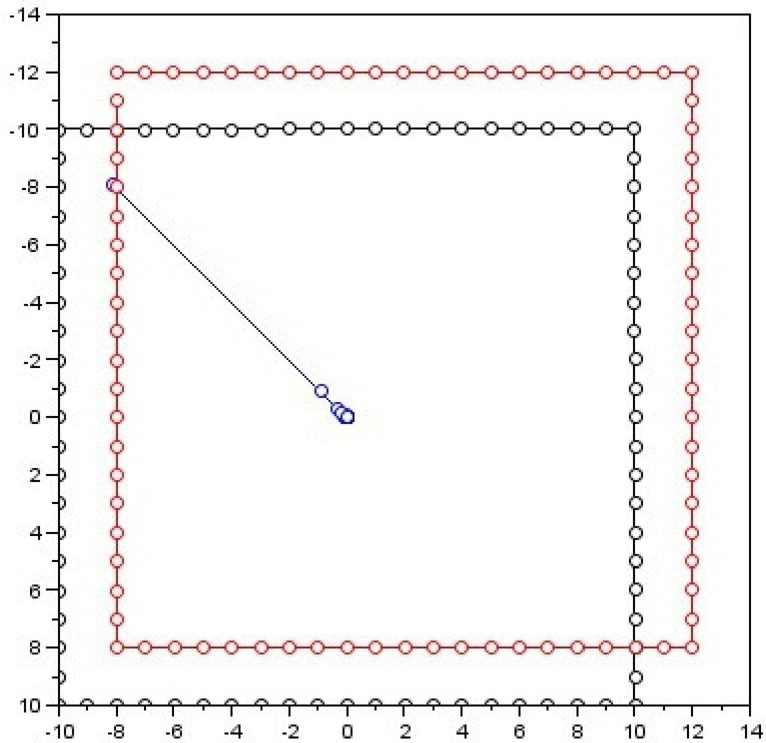

Figure 2 : influence du barycentre sur un contour d'un carré.

2. Filtrage des descripteurs de Fourier

Dans cette deuxième partir nous allons voir les avantages et les inconvénients d'utiliser la méthode des descripteurs de Fourier sur différents contour.

Tout d'abord, comme on l'a vu précédemment, un des avantages des descripteurs de Fourier se fait sur des figures non rectilignes. En effet, pour une forme sphérique (voir figure 1), on peut éliminer certains descripteurs tout en gardant un contour assez représentative de la forme initiale. Du coup, la signature du contour est plus compacte.

Malheureusement, cette méthode n'est pas optimale pour des contours rectilignes. Comme le montre la figure 3 ci dessous.

Pour annuler certains descripteurs de Fourier voici la fonction.

```
// annule certains des descriteurs de Fourier
function annule=rdfAnnuleDescFourier(desc, ratio)
    annule = desc;
    if ratio == 0 then
        annule = zeros(size(desc,1),1);
    else
        nbValSup = size(desc,1)*ratio;
        annule = desc(1:nbValSup);
    end
endfunction
```

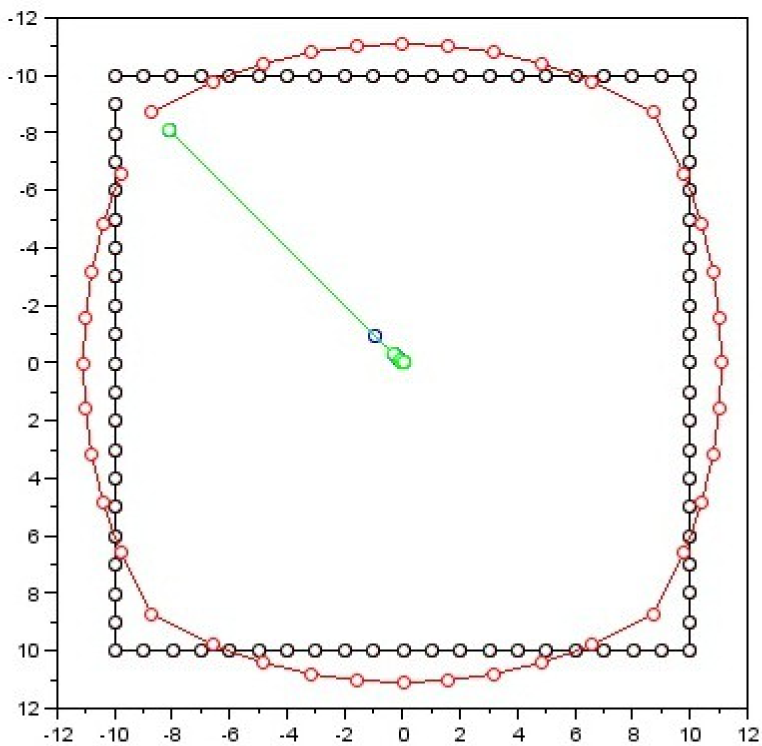

Figure 3 : représentation d'un contour d'un carré avec la réduction du nombre de descripteurs.

Nous verrons dans la dernière partie plusieurs cas de contour de forme où la méthode des descripteurs de Fourier est plus ou moins optimales.

3. Réduction d'une chaîne de contour

Dans cette troisième partie, nous allons découvrir et utiliser l'algorithme de la corde sur différents contours. L'algorithme de la corde permet d'éliminer des points d'un contour en l'approchant par une série de segments de droite. On peut tout de suite dire que cette méthode sera plus représentative du contour pour une forme rectiligne, en éliminant les points se trouvant sur un même segment.

On peut constater dans la figure 4 ci dessous, que la distance maximale du contour initial et du contour obtenu n'influence pas la représentation. On peut donc définir une distance maximale optimale pour les formes rectilignes et qui demande le moins de calcul, tout en représentant au mieux le contour. Voici le bout de code pour l'implémentation de l'algorithme de corde :

```
if (real(debut)-real(fin))==0 then
    d = abs(real(cont)-real(debut));
else
    if (imag(debut)-imag(fin))==0 then
        d = abs(imag(cont)-imag(debut));
    else
        a = (imag(debut)-imag(fin))/(real(debut)-real(fin));
        b = imag(fin) - a*real(fin);
        d = (abs(real(cont)*a - imag(cont) +b))/(sqrt(1 + a*a));
    end
end
```

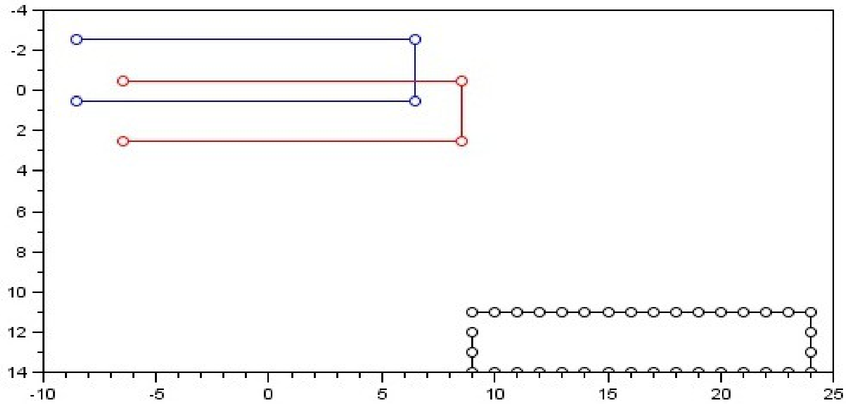

Figure 4 : représentations d'un contour d'un rectangle avec l'algorithme de la corde.

Voici la représentation graphique avec le contour initial (rouge), le contour avec une distance maximale de 1 pixel (vert) et enfin celui avec une distance maximale de 0,5 pixel (bleu). On constate par contre que sur un contour non rectiligne, la représentation devient de plus en plus polygonale.

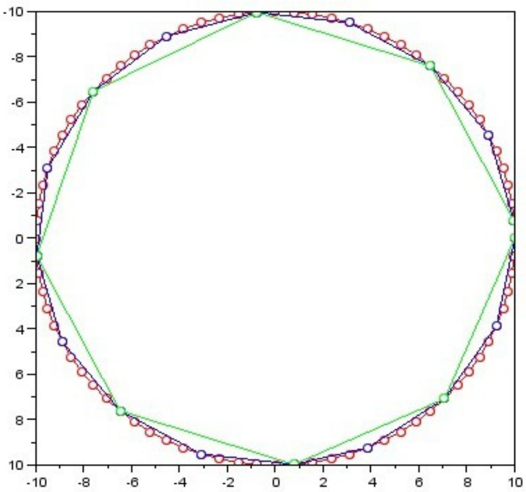

Figure 5 : représentations des contours d'un cercle.

4. Comparaison des deux approches

Dans cette quatrième et dernière partie, nous allons comparer les 2 méthodes, en distinguant les avantages et les inconvénients de l'un l'autre. Nous avons déjà constater que les descripteurs de Fourier donnaient une meilleure représentation pour les formes sphériques tout en aillant limité le nombre de coefficients. De son coté l'algorithme de la corde permettaient une représentation plus réaliste des formes rectilignes en limitant des calculs inutiles. Mais qu'en est il pour des formes peu communes, comme par exemple une croix ou une forme non définie ?

Pour une meilleure visibilité des contours, nous avons décidé de décaler le barycentre, des représentations obtenues par les méthodes. Voici, les images de ces 2 formes. En noir le contour initial, en rouge celui obtenu par les Descripteurs de Fourier et enfin le bleu, celui de la méthode de l'algorithme de la corde.

Pour la croix, en réduisant le nombre de coefficient de Fourier, on peut constater que le contour est différent de la croix initiale, mais représente toute fois une croix. Avec l'algorithme de la corde, la visualisation est directe car la croix est formée de ligne.

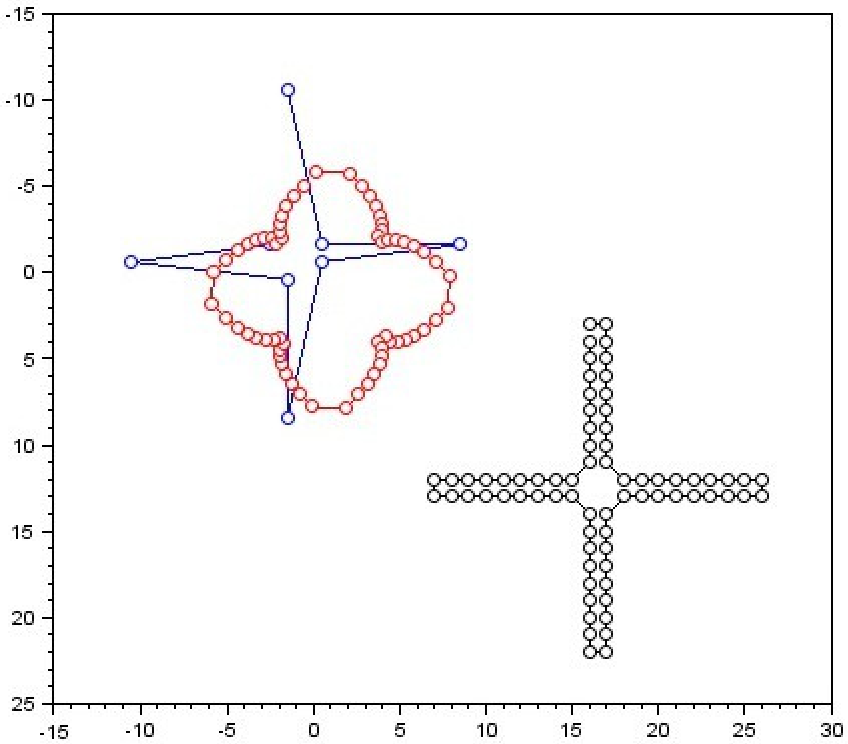

Figure 6 : représentations des contours d'une croix.

Pour la patate ou une forme quelconque (figure 7), l'algorithme de la corde (en bleu) se rapproche trop des points initiales et ne supprime pas les points inutiles pour reconnaître la forme (par exemple : le pic à droite sur le contour). Par contre, les descripteurs de Fourier permettent avec une signature compacte de représenter le contour de la forme en supprimant les points inutiles. On s’aperçoit donc que la forme quelconque se rapproche d'un cercle.

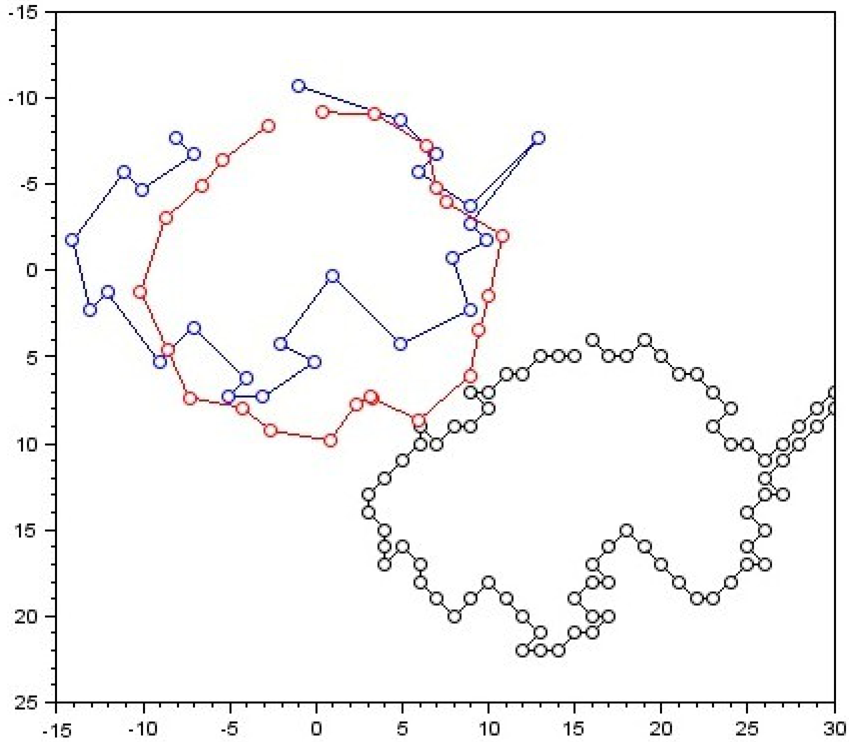

Figure 7 : représentations des contours pour une forme quelconque.

Conclusion :

Les 2 démarches pour approcher un contour d'une forme sont intéressantes. L'algorithme de la corde est très utile pour des formes rectilignes comme le triangle ou la croix, car elle permet de réduire le nombre de point, tout en gardant la forme de la figure. Cependant pour les formes comme la patate, cet algorithme prend en compte des points extrêmes et non représentatives de la figure (la pointe en bleu sur la figure ci-dessus). Par contre la méthode des descripteurs de Fourier est meilleure pour ces formes car elle supprime justement ces points et est médiocres avec des formes rectilignes.