Damien Druel

TP 2 : Contours d’une forme

Introduction

Dans ce TP, nous allons dans un premier temps comment sont représentés les contours d’une forme et comment afficher ce contour avec le langage R.

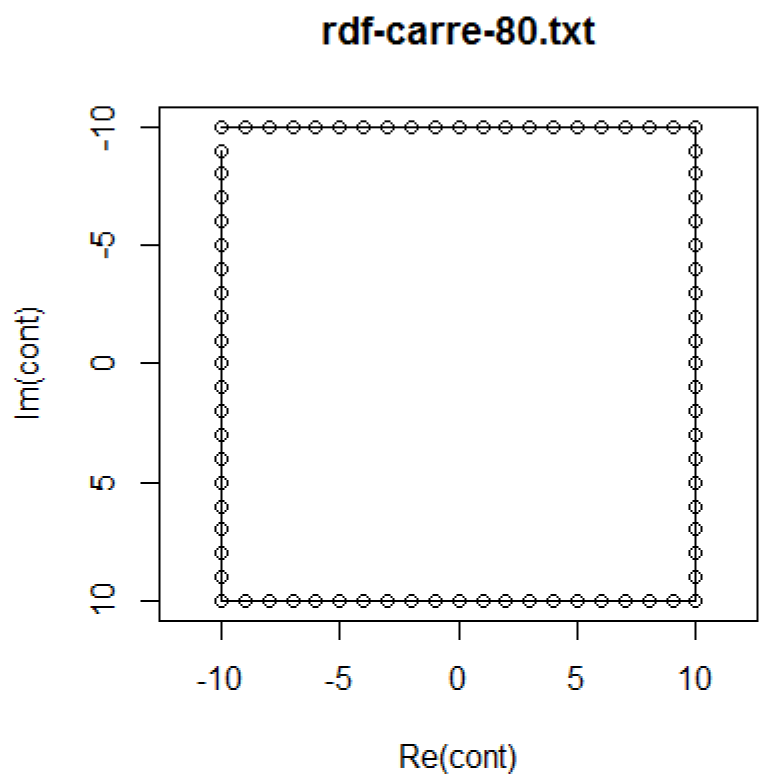
Ensuite nous verrons la méthode des descripteurs de Fourier, ainsi que le filtrage de ces descripteurs pour pouvoir garder une forme d’un contour tout en réduisant le nombre de points élémentaires de ce contour.

Puis nous verrons la méthode de la corde qui permet aussi de réduire le nombre de point d’un contour en faisant une réduction de chaîne élémentaire.

Enfin, nous finirions par comparer ces deux méthodes sur différentes images/formes pour en déduire quelle méthode est la plus efficace sur quel type de forme.

Damien Druel

1. Code R



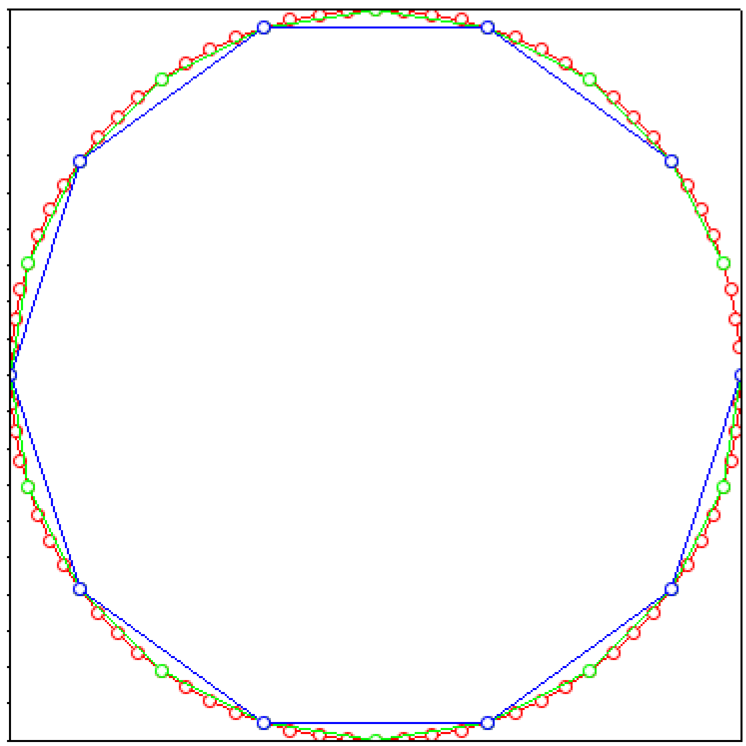
Comment sont codés les points du contour dans la variable cont? Quel est l'intérêt du dernier argument de la méthode plot?

- Les points sont codés par une liste de Tuples de coordonnés en X et en Y pour chaque points du carré.

- Le parametre “ ylim = rev (range (Im (cont)))) “ prend la liste des points, trouve le min et le max et s’en sert pour faire les valeurs affichés sur la ligne d’abscisse et d’ordonnée, ici { -10, 10 }

Damien Druel

Modifier le code du fichier rdfTesteContours.R afin d'afficher en rouge le contour d'un cercle. En ne conservant qu'un point sur 4, puis un point sur 8, constituer deux autres contours approchant la forme circulaire et les afficher respectivement en bleu et en vert dans la même fenêtre.



2. Descripteur de Fourier

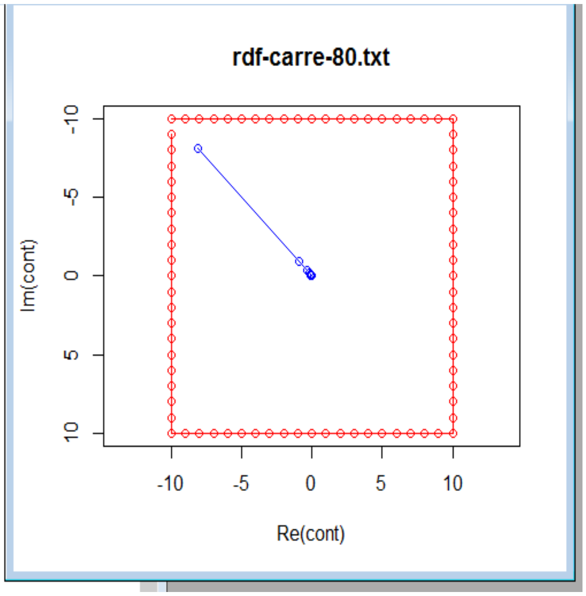
2.1. Descripteurs de Fourier

Calculer les descripteurs de Fourier du contour contenu dans le fichier rdf-carre-80.txt en utilisant la fonctionfft de R, normalisée en divisant par le nombre de points. Vérifier qu'une transformée de Fourier inverse permet de reconstituer exactement le contour initial.

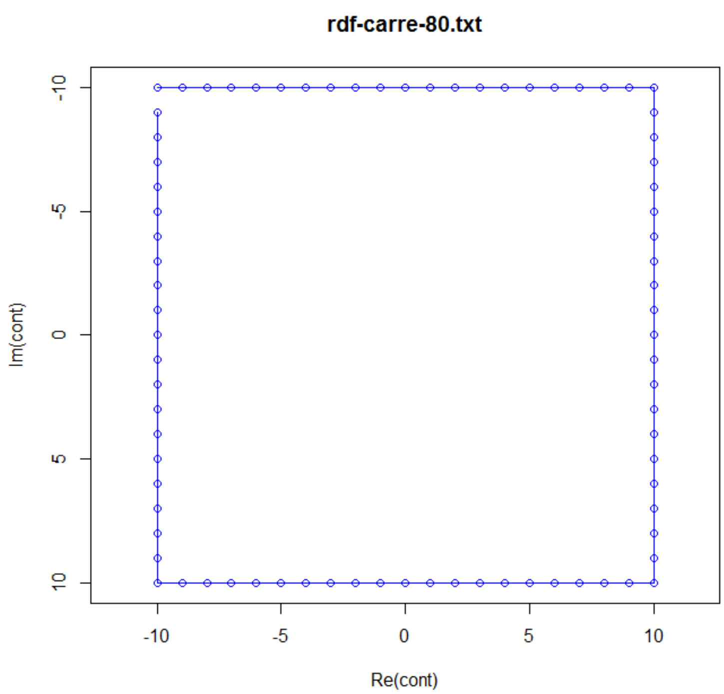
Damien Druel

Descripteur de fourier :

```
> desc
 [1]  0.00000000+0.00000000i -8.10986264-8.10986264i  0.00000000+0.00000000i
 [4]  0.00000000+0.00000000i  0.00000000+0.00000000i -0.32842678-0.32842678i
 [7]  0.00000000+0.00000000i  0.00000000+0.00000000i  0.00000000+0.00000000i
[10] -0.10434317-0.10434317i  0.00000000+0.00000000i  0.00000000+0.00000000i
[13]  0.00000000+0.00000000i -0.05235586-0.05235586i  0.00000000+0.00000000i
[16]  0.00000000+0.00000000i  0.00000000+0.00000000i -0.03261346-0.03261346i
[19]  0.00000000+0.00000000i  0.00000000+0.00000000i  0.00000000+0.00000000i
[22] -0.02318122-0.02318122i  0.00000000+0.00000000i  0.00000000+0.00000000i
[25]  0.00000000+0.00000000i -0.01808078-0.01808078i  0.00000000+0.00000000i
[28]  0.00000000+0.00000000i  0.00000000+0.00000000i -0.01515659-0.01515659i
[31]  0.00000000+0.00000000i  0.00000000+0.00000000i  0.00000000+0.00000000i
[34] -0.01349426-0.01349426i  0.00000000+0.00000000i  0.00000000+0.00000000i
[37]  0.00000000+0.00000000i -0.01267511-0.01267511i  0.00000000+0.00000000i
[40]  0.00000000+0.00000000i  0.00000000+0.00000000i -0.01251930-0.01251930i
[43]  0.00000000+0.00000000i  0.00000000+0.00000000i  0.00000000+0.00000000i
[46] -0.01299458-0.01299458i  0.00000000+0.00000000i  0.00000000+0.00000000i
[49]  0.00000000+0.00000000i -0.01420127-0.01420127i  0.00000000+0.00000000i
[52]  0.00000000+0.00000000i  0.00000000+0.00000000i -0.01642038-0.01642038i
[55]  0.00000000+0.00000000i  0.00000000+0.00000000i  0.00000000+0.00000000i
[58] -0.02026843-0.02026843i  0.00000000+0.00000000i  0.00000000+0.00000000i
[61]  0.00000000+0.00000000i -0.02712848-0.02712848i  0.00000000+0.00000000i
[64]  0.00000000+0.00000000i  0.00000000+0.00000000i -0.04049786-0.04049786i
[67]  0.00000000+0.00000000i  0.00000000+0.00000000i  0.00000000+0.00000000i
[70] -0.07131611-0.07131611i  0.00000000+0.00000000i  0.00000000+0.00000000i
[73]  0.00000000+0.00000000i -0.16965274-0.16965274i  0.00000000+0.00000000i
[76]  0.00000000+0.00000000i  0.00000000+0.00000000i -0.90481100-0.90481100i
[79]  0.00000000+0.00000000i  0.00000000+0.00000000i
```



Inverse :



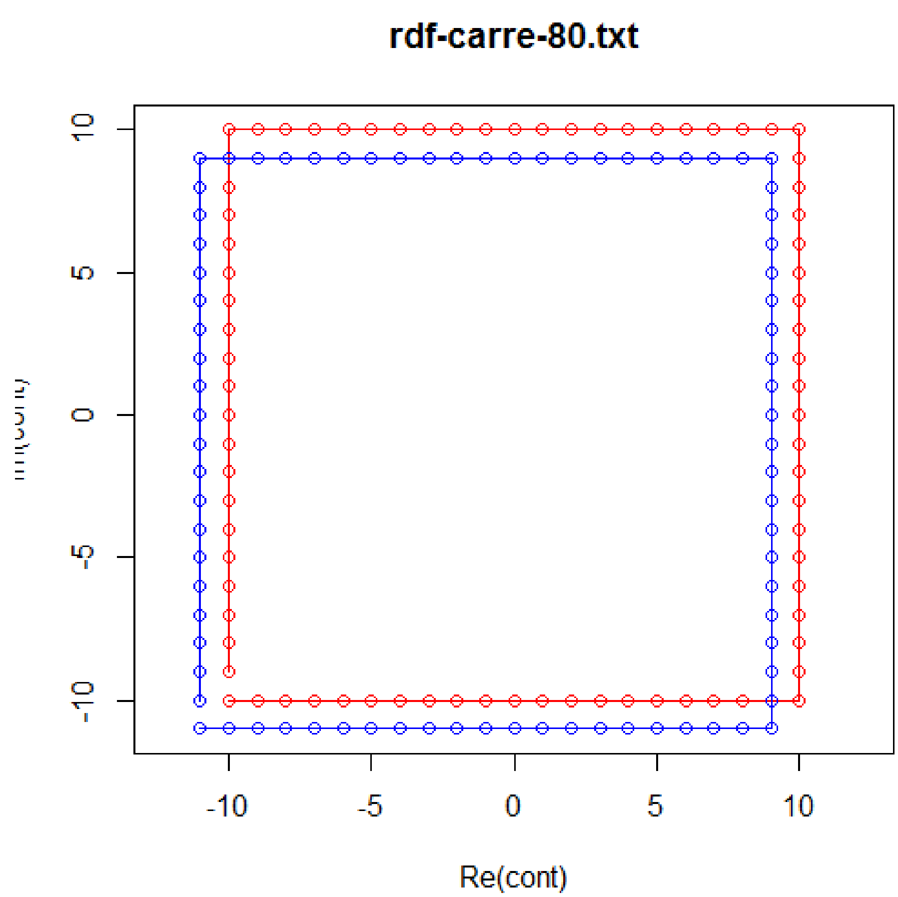
La transformée de Fourier inverse permet bien de retrouver les points exacts du contour du carré.

Damien Druel

A quoi correspond le descripteur de Fourier Z_0 d'une forme décrite par son contour? Vérifier son influence en additionnant une constante complexe au descripteur Z_0 avant de reconstruire le contour par transformation inverse.

Le descripteur de Fourier Z0 est le barycentre de la forme. Si on modifie la valeur de Z0 dans la matrice de descripteurs avant l'inversion, en lui additionnant une constante complexe (par exemple de -1 + -1i), cela aura pour effet de déplacer le barycentre et donc de translater la forme.

voici l'exemple en question :

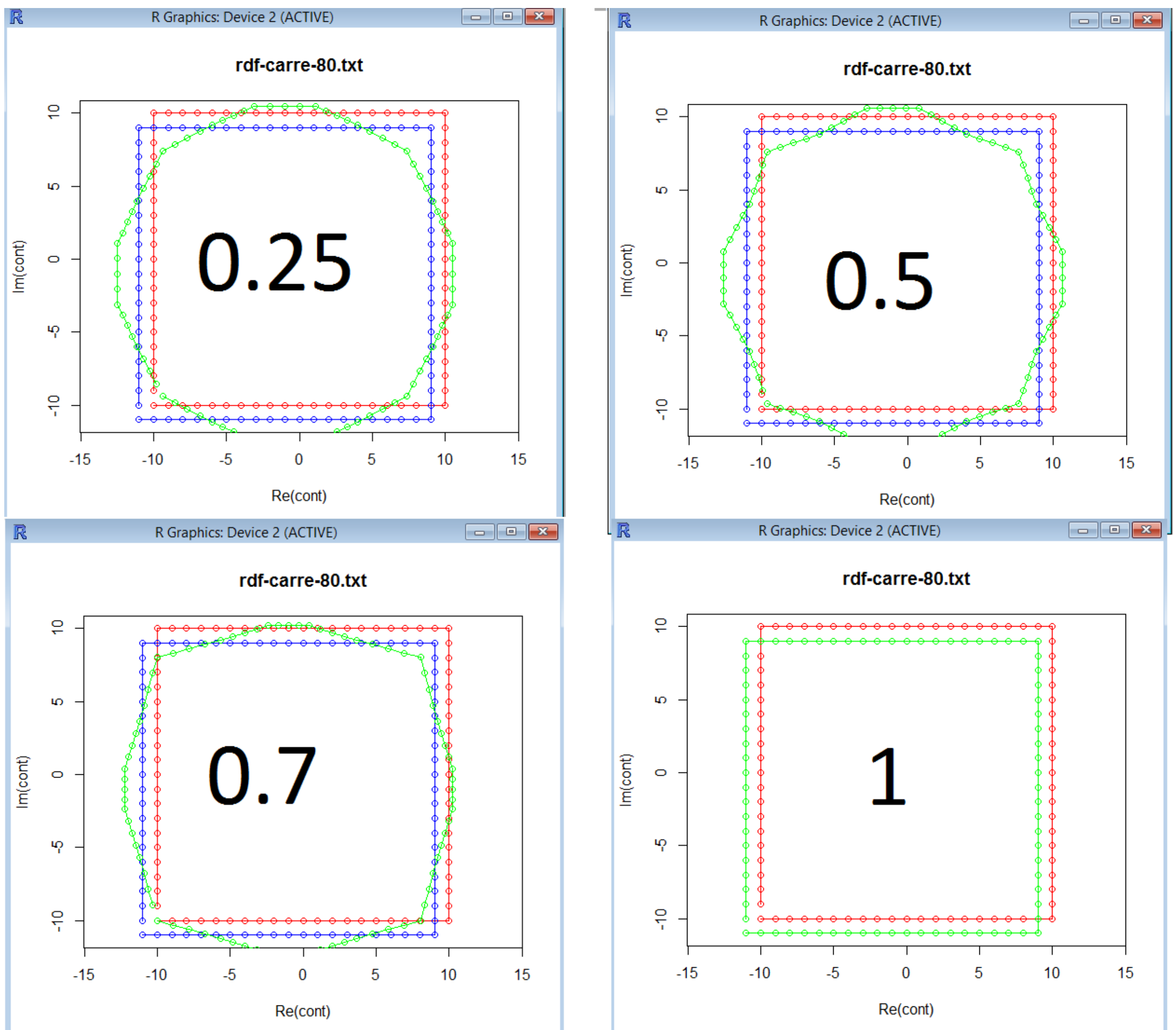


On voit bien que la forme a translaté de -1 en abscisse et -1 en ordonnée.

2.2. Filtrage des descripteurs de Fourier

Utiliser la fonction **rdfAnnuleDescFourier** sur les descripteurs calculés sur le carré dont le contour comporte 80 points. Vérifier que le contour reconstitué se **simplifie** quand on augmente le nombre de descripteurs annulés, c'est à dire quand on rend la signature plus **compacte**.

rouge = carré avant translation
bleu = inverseFourier
vert = inverseFourier(rdfAnnulDescFourier(x))



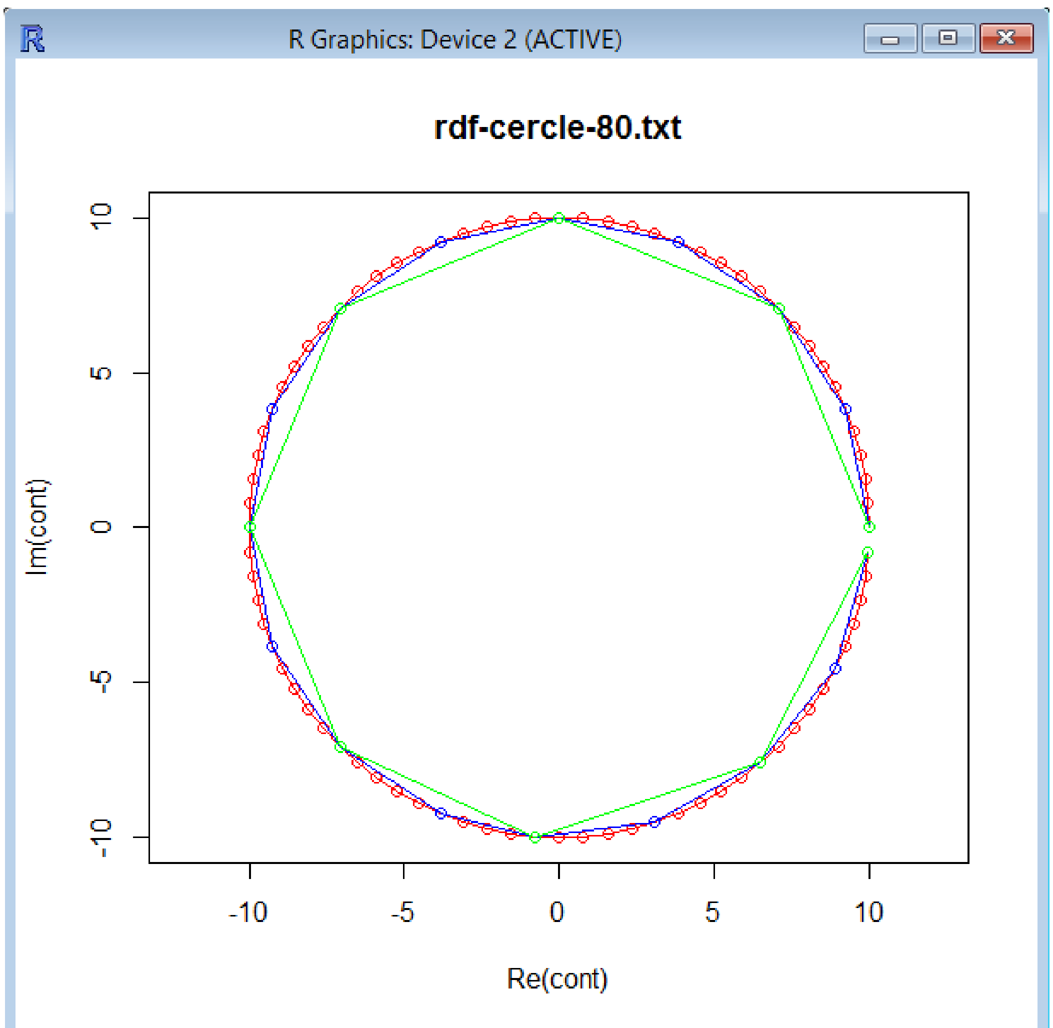
On voit bien que le contour reconstitué se simplifie .

Damien Druel

3. Réduction d'une chaîne de contour

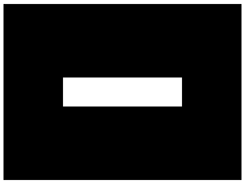
réduire le contour décrivant un cercle (fichiercercle-80.txt) en utilisant une distance maximale d'abord de 0.5 pixel, puis de 1 pixel. Afficher les trois contours (initial, avec dmax=0.5 et avec dmax=1) avec trois couleurs différentes dans une même figure.

rouge : cercle
bleu : dmax = 0.5
vert : dmax = 1

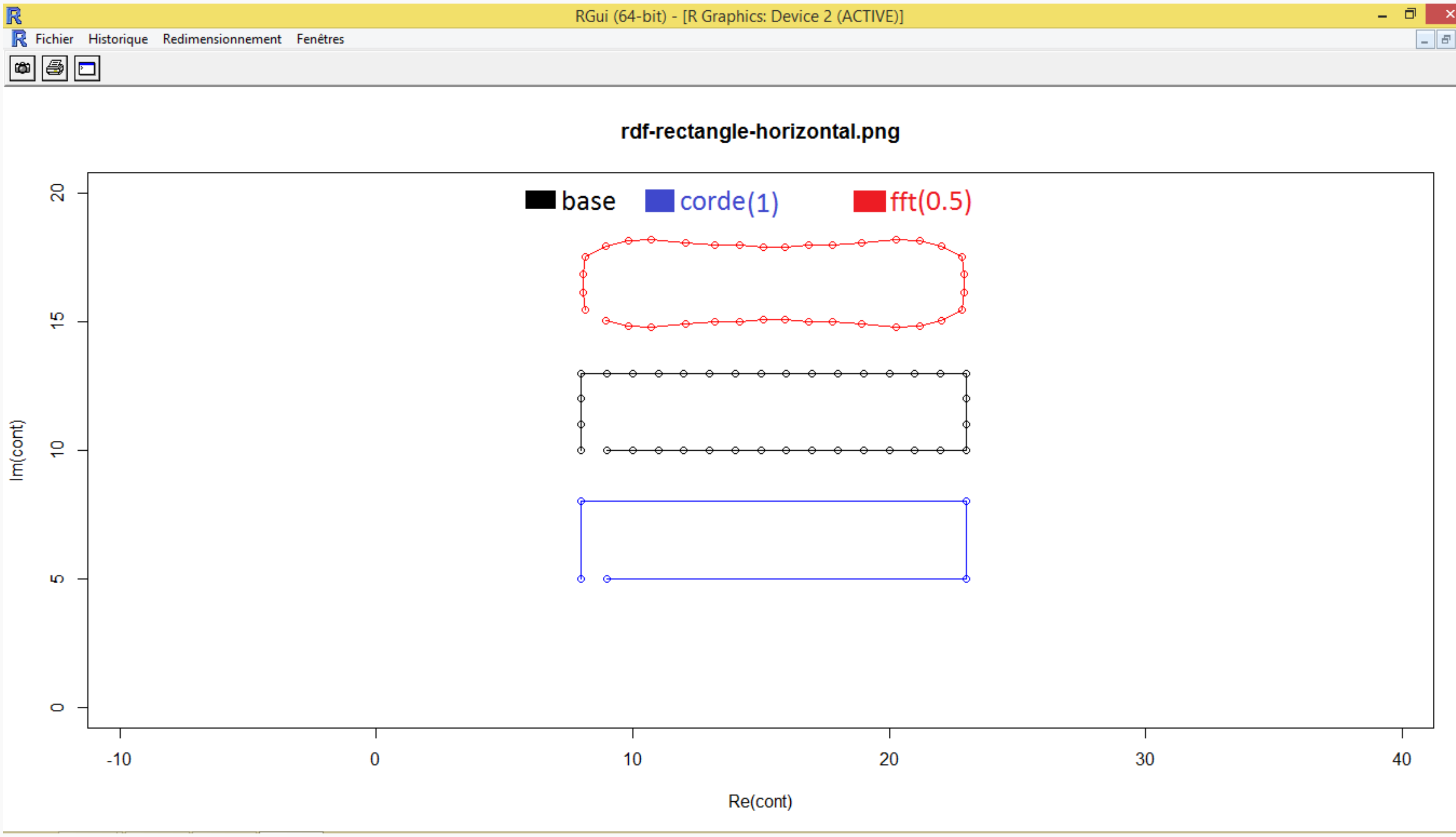


Damien Druel

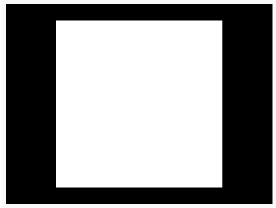
4. Comparaison des deux approches



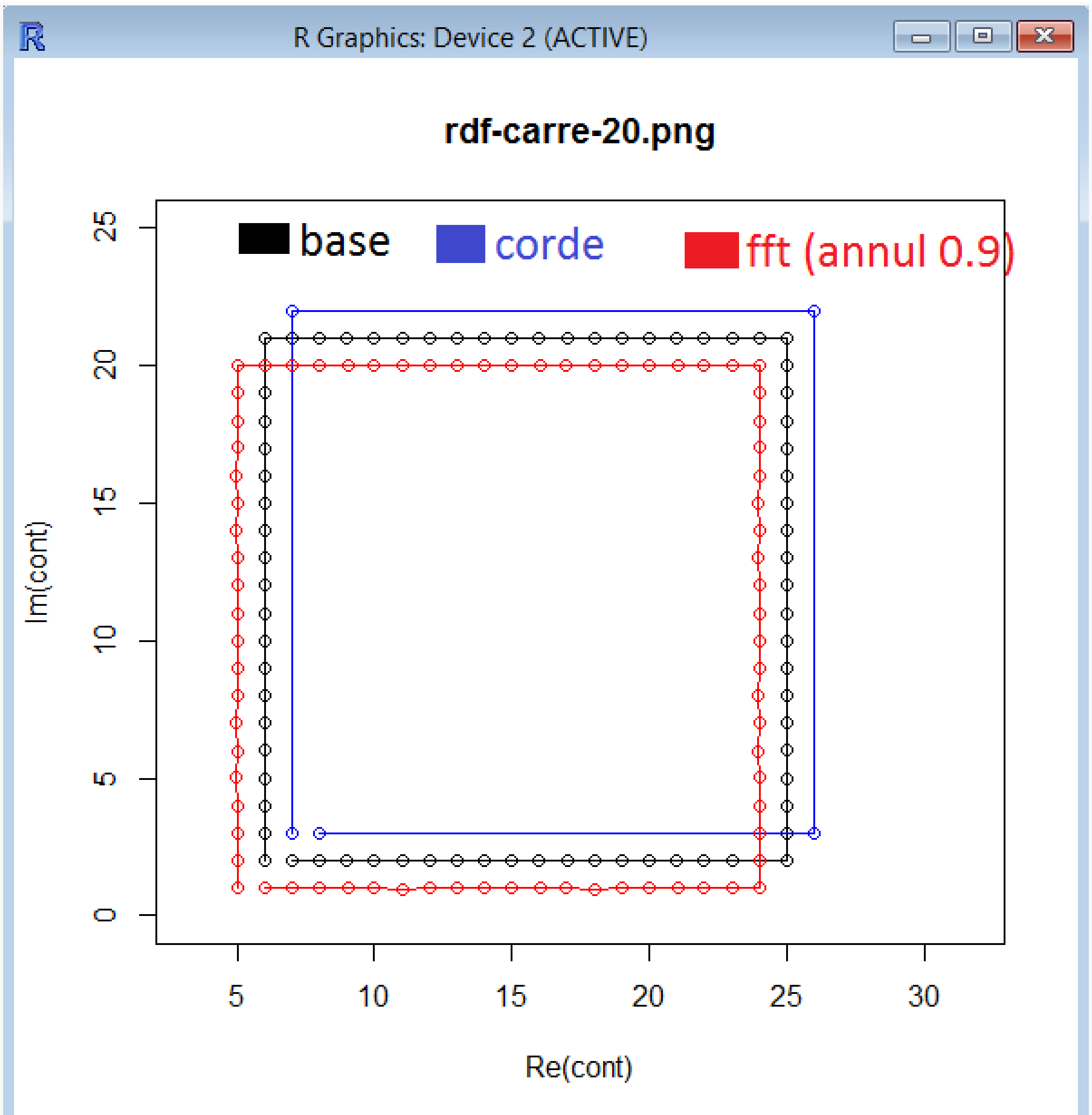
Rectangle H.



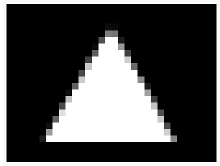
Damien Druel



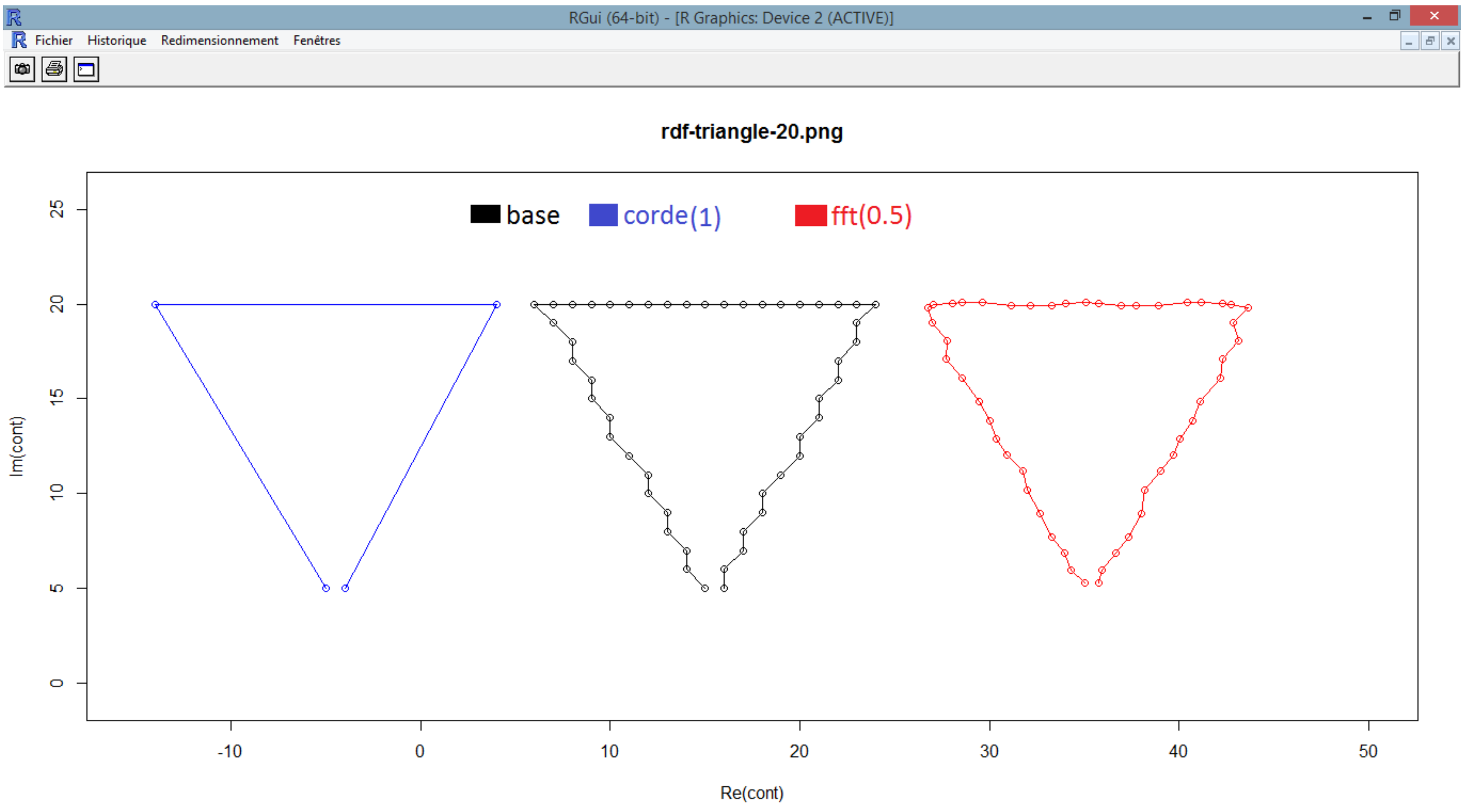
Carré



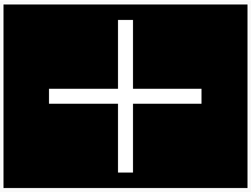
Damien Druel



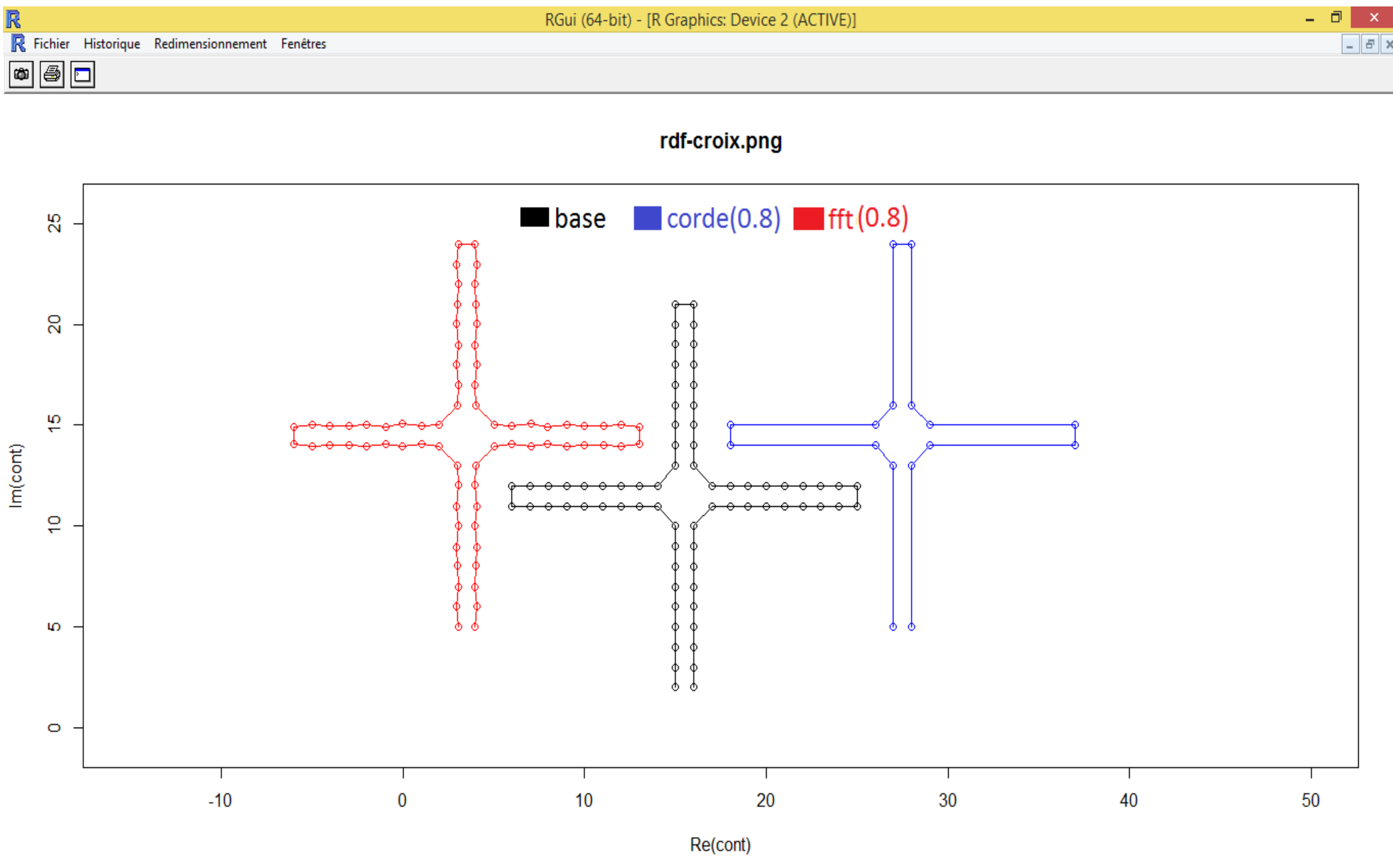
Triangle



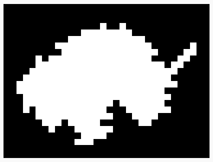
Damien Druel



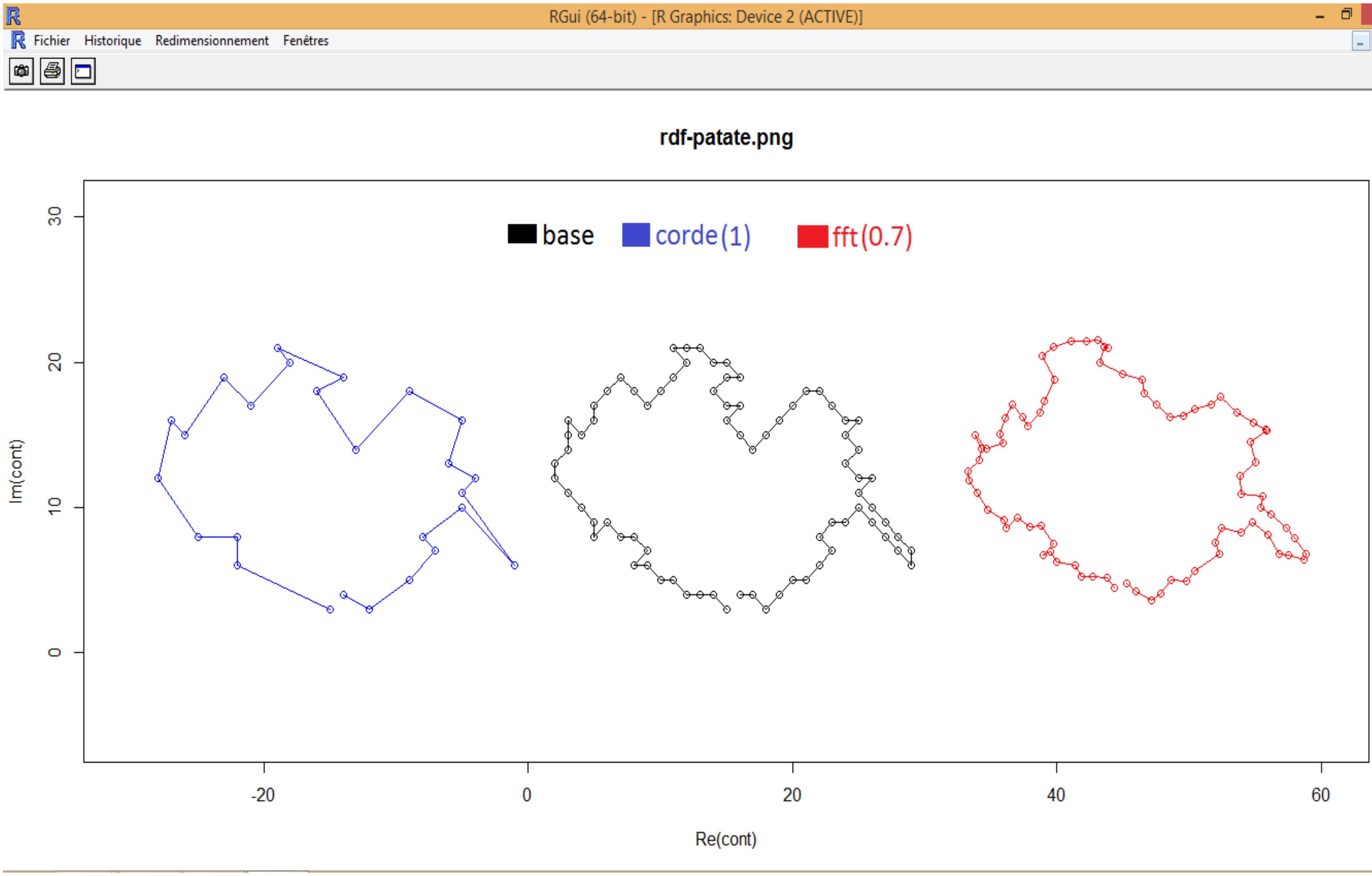
Croix



Damien Druel



Patatoïde



- Pour le rectangle, le carré, le triangle et la croix, on voit clairement que l’algorithme des cordes peut réduire le contour en quelques points extrêmes, alors que Fourier ne peut réduire le contour qu’en 50-90% des points si on veut garder la forme originale. Ceci est dû au fait que ces quatre formes géométriques sont essentiellement composés de longues lignes droites, qui, auparavant représentées par une dizaine points peuvent en fait être représentées par deux points.

- Pour la patate, on voit plutôt que Fourier parvient mieux à garder la forme “naturelle” de la patate, avec des arrondissement, alors que l’algorithme de corde n’y parvient pas. Ceci est du au fait que la patate n’est pas composé de lignes droites mais d’un ensemble de petits arrondissements que l’algorithme de corde est incapable de conserver.

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 représentatifs de la figure (la pointe en bleu sur la figure de la patate). Par contre la méthode des descripteurs de Fourier est meilleure pour ces formes car elle supprime justement ces points et est médiocres avec les formes rectilignes.