

# Implémentation de l'algorithme Inpainting

Auteur : Hobeau BAE  
Contact : b845548@gmail.com

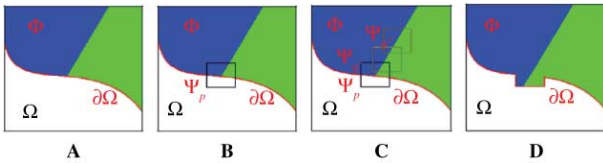
## I. INTRODUCTION

**L**E but de ce projet est d'implémenter un algorithme qui permet de supprimer un objet qu'on sélectionne dans une image, et de reconstruire des pixels qui ont été supprimés.

## II. DÉVELOPPEMENT

Premièrement on sélectionne une partie d'image et de la remplir par le pixel vide. Dans ce cas il peut avoir un problème est que cette valeur peut déjà être présente dans l'image. Pour cela, on choisit une valeur qui n'est pas présente dans l'image. Comme on a rempli le pixel spécifique, on peut détecter le contour de la zone supprimée. Le principe de cet algorithme est de chercher une zone de pixels qui se ressemble au contour des pixels supprimés. Si on cherche sur tous les pixels, on tombe souvent sur la même zone, cela construit un pattern répétitif, pour cela on limite le champ de recherche autour du contour. Et puis on calcule la somme de la distance entre une zone de pixels et le contour de pixels, on itère sur le champ de recherche. Finalement on remplace des pixels vides par ceux de la zone sélectionnée par la plus petite distance avec le contour de pixels.

## III. SCHEMA



## IV. ALGORITHM

```
img_original = lire()
img = remplir_vide(img_original)

tant que img contient un pixel vide
    contour_xy = chercher_contour(img)
    contours = decouper(contour_xy)
    probables = similaire(img, contours)
    img = remplacer(img, probables, contours)
```

## V. FORMULE

$$\begin{aligned} distance(pixel1, pixel2) = & (pixel1_r - pixel2_r)^2 \\ & + (pixel1_g - pixel2_g)^2 \\ & + (pixel1_b - pixel2_b)^2 \end{aligned} \quad (1)$$

$$similarite = \frac{\sum_{i=1}^n distance(zone_i, contour_i)}{n} \quad (2)$$

## VI. RÉSULTAT OBTENU



(1) image original



(2) image reconstruite

## VII. CONCLUSION

On perçoit qu'il y a une cohérence de l'ensemble de pixels construits, sauf le programme sélectionne le contour de pixel en diagonal vers le bas droit, cela produit un motif qui se répète en diagonal. Ce problème peut être résolu en partant de tous les directions chercher le contour.