Equation de Poisson et traitement d'image

Groupe de travail thématique

4TQMS801S

# Equation de Poisson et traitement d'image

Virginie MONTALIBET

Sébastien Eyzat

April 18, 2020

# Contents

# 1 Présentation du problème

Le traitement d'images est un ensemble de méthodes permettant d'étudier et de transformer une ou plusieurs images à l'aide de moyens mathématiques et numériques. Le principe du traitement d'images consiste à extraire certaines informations de celles-ci, afin de les étudier ou de les modifier.Il est utilisé dans beaucoup d'applications telles que l'amélioration du contraste, l'application d'un filtre(flou, lissage, changement de couleurs), ou encore les détections et identifications d'objets par exemple.

Aujourd'hui, nous nous intéressons à l'incrustation d'images. A partir de deux images, comment sélectionner une partie de la première et l'incruster de la manière la plus naturelle possible dans la seconde ?
Afin d'éclaircir nos propos et d'identifier les problèmes que nous devons résoudre, voici un exemple de ce que nous souhaitons faire.
Nous disposons des deux images présentées ci-dessous, l'image T(arget) et l'image S(ource).

Figure 1: Image T

Figure 2: Image S

L'objectif est d'incruster toute ou partie de l'image S dans l'image T. En terme de manipulations, cela consiste à effectuer un copier/coller ou encore un clônage de la seconde image dans la première. Le résultat que nous attendons pour une incrustation "réussie" est un résultat comme celui présenté ci-dessous :

Figure 3: Image finale attendue

Cette image est extraite d'une simulation effectuée sur 'ipol.im/...'.Nous comparerons les images obtenues avec nos algorithmes avec celle-ci à la fin de ce rapport.

**Remarques** Ce résultat semble naturel, les frontières entre l'image collée et l'arrière plan sont très peu visibles, les oiseaux semblent faire partie de l'image finale. Mais comment obtenir un tel résultat ?
Reprenons nos deux images séparées, T et S. Commençons par effectuer un simple copier/coller. Certains pixels de T sont donc écrasés par ceux de S. En effectuant cette manipulation voici l'image que nous devrions obtenir :

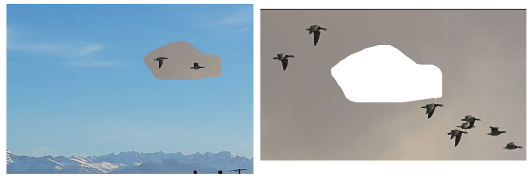


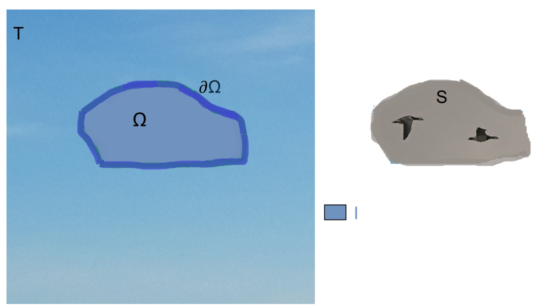
Figure 4: Simple copier/coller

Ce résultat n'est bien entendu pas convenable et bien loin de l'image finale attendue. Le découpage et le collage entre l'arrière-plan et l'image "objet" sont beaucoup trop visibles, les couleurs ne sont pas les mêmes et incohérentes. Nous souhaitons obtenir un résultat beaucoup plus naturel, comme écrit plus haut. Cette simple manipulation n'est donc pas suffisante pour effectuer le clônage cohérent d'une image dans une autre.

Il semble évident de vouloir modifier l'image finale obtenue afin qu'elle paraisse la plus naturelle possible. Nous verrons tout au long de ce rapport, quels sont les changements a effectuer, et comment la résolution d'une équation aux dérivées partielles, nous permet d'obtenir un bien meilleur résultat.

## 1.1 Traduction du problème sous formes mathématiques.

Comme écrit plus haut nous souhaitons apporter des modifications au précédent collage afin qu'il corresponde au mieux à nos attentes. Pour ce faire, considérons I, la partie de l'image finale à modifier. Nous pouvons ainsi représenter le problème sous forme schématique.




Ici :

- T est l'image "Target", l'image destination, l'image sur laquelle s'effectuera le collage, l'arrière plan.
- S est l'image "Source", l'image que nous souhaitons coller.
- $\Omega$ est le domaine dans lequel se trouvent nos inconnues.
- $\partial\Omega$ est la frontière de $\Omega$.
- I est notre inconnue, la partie de l'image que nous ne connaissons pas et que nous voulons remplir.

Nous souhaitons donc trouver une fonction I qui satisfasse un certain nombre de critères, afin de correspondre au résultat voulu. Cette fonction représente notre image modifiée. Mais quelles sont les conditions qu'elle doit remplir pour que le rendu soit le meilleur possible? Notre nouvelle image I doit-elle être plus proche de l'image collée S, ou de l'image d'arrière-plan T ?
Pour répondre à cette question, reprenons le problème de départ.

Pour que l'image obtenue paraisse naturelle,il faut que celle-ci dénature le moins possible les deux images sélectionnées au départ. En effet, nous devons garder les détails de l'image que nous voulons coller, S, ne pas modifier les variations qu'elle pourrait posséder comme les contours des objets lui appartenant, par exemple.Nous devons pouvoir retrouver les informations présentes dans celle-ci. Il faut donc que les variations présentes dans $I$ soient presque identiques à celles de S. Mathématiquement, de manière équivalente, nous cherchons une fonction I dont le gradient est le plus proche possible, ou identique à celui de l'image collée, S. Nous cherchons donc :

$$min \iint_{\Omega} ||\nabla I_{x,y} - \nabla S_{x,y}||^2 dxdy$$

Mais les démarcations entre notre image collée et l'image d'arrière-plan T, ne doivent pas non plus être visibles, il faut donc que les pixels se situant sur cette partie là, i.e $\partial\Omega$, soient le plus proches possible de T . Nous voulons donc, mathématiquement, chercher la fonction I qui vérifie:

$$I_{(x,y)} = T_{x,y} \ sur \ \partial\Omega$$

Répondre au problème implique donc de résoudre un problème variationnel classique auquel des conditions sur le bord de Dirichlet sont ajoutées. Réécrivons le problème mathématique que nous cherchons à résoudre :

$$\begin{cases} min \iint_{\Omega} ||\nabla I_{x,y} - \nabla S_{x,y}||^2 dxdy \\ \qquad I_{(x,y)} = T_{x,y} \ sur \ \partial\Omega \end{cases}$$

## 1.2 Résolution

Pour résoudre cette équation, nous devons trouver le minimum de la fonction g suivante :

$$g(I) = \int_{\Omega} ||\nabla I(x) - v(x)||^2 dx \tag{1}$$

Si g admet un minimum alors, celui-ci annule son gradient.

**Calcul du gradient de g**

$$g(I + \epsilon u) - g(I) = \int_{\Omega} ||\nabla(I + \epsilon u) - v||^2 - ||\nabla I - v||^2 dx$$

En développant :

$$\begin{aligned} g(I + \epsilon u) - g(I) &= \int_{\Omega} ||\nabla I - v||^2 + ||\nabla \epsilon u||^2 + < 2(\nabla I - v), \epsilon \nabla u > - ||\nabla I - v||^2 dx \\ &= \int_{\Omega} \epsilon^2 ||\nabla u||^2 + < 2(\nabla I - v), \epsilon \nabla u > dx \\ &= 2 \int_{\Omega} (\nabla I - v) \times (\nabla \epsilon u) + O(\epsilon^2) \\ &= -2\epsilon \int_{\Omega} < \nabla I - v, \nabla u > + O(\epsilon^2) \\ &= -2 \int_{\Omega} < \nabla I - v, \nabla u > + O(\epsilon) \\ &= 2 < div(\nabla I - v), u > + O(\epsilon) \end{aligned}$$

Par identification, le gradient de g vaut

$$\nabla g(I) = (\Delta I - div(v))$$

Le minimum de g annule son gradient donc on cherche I tel que :

$$0 = (\Delta I - div(v))$$

Ici v = $\nabla S$

$$\Delta I = div(\nabla S)$$
$$\Delta I = \Delta S$$

Trouver le minimum de la fonction revient donc à résoudre l'équation :

$$\Delta I = \Delta S$$

Cette équation est l'équation de Poisson.

Résoudre ce problème variationnel est équivalent à résoudre l'équation de poisson avec conditions aux bords de Dirichlet, suivante :

$$\begin{cases} \Delta I = \Delta S \ sur \ \Omega \\ I = T \ sur \ \partial\Omega \end{cases}$$

Nous verrons deux approches afin de résoudre cette équation, la première utilisant des discrétisations, la seconde la méthode de Fourier.

# 2 Résolution

Nous verrons dans un premier temps, comment résoudre cette équation à l'aide de discrétisation et de différences finies. Puis nous résoudrons celle-ci à l'aide des transformées de Fourier. Rappelons que nous souhaitons résoudre l'équation de Poisson suivante :

$$\begin{cases} \Delta I = \Delta S \ sur \ \Omega \\ I = T \ sur \ \partial\Omega \end{cases}$$

## 2.1 Discrétisation

Avant de discrétiser les Laplaciens de nos deux images, il faut définir le maillage que nous allons utiliser.

### 2.1.1 Qu'est qu'une image

Une image peut être représentée comme une succession de pixels. En traitement d'image, ce sont d'ailleurs sur ces pixels que le traitement est effectué. Leur modification entraîne la modification de l'image globale.Nous verrons donc une image comme la succession de pixels et nous pouvons la représenter comme une grille, dans laquelle chaque carré représente un pixel. Schématiquement, nous pourrions donc découper notre image comme ci-dessous Nous

Figure 5: Maillage d'une image

numéroterons les pixels suivant la règle suivante. Le premier pixel est situé en haut en gauche, puis il suffit de parcourir la grille comme ci-dessous :

Les pas d'espaces sont donc égaux et valent 1. Dans la suite nous considérons que notre image est de taille $N \times M$.

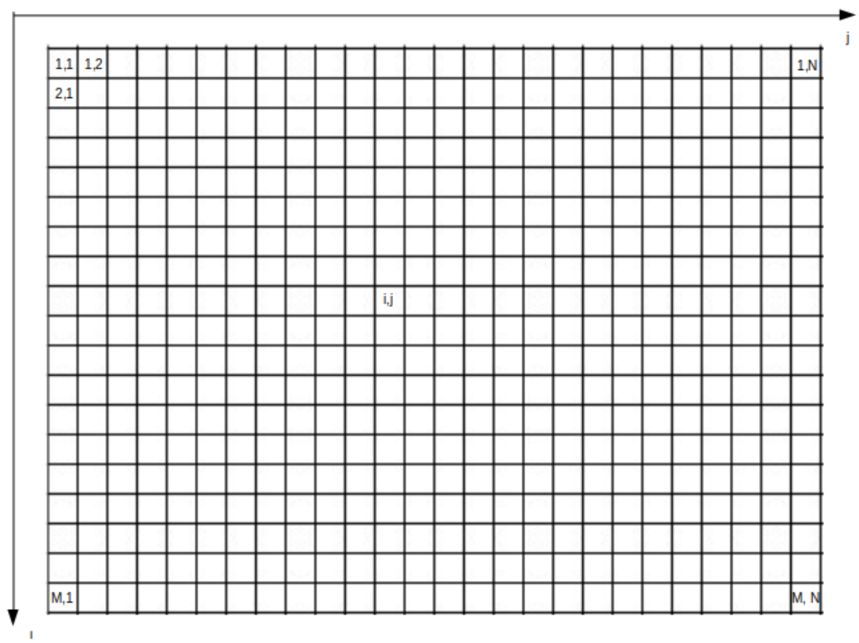



Figure 6: Parcours d'une grille

## 2.2 Les différents opérateurs nécessaires

Nous utiliserons certains opérateurs que nous définissons ici afin de ne pas alourdir les différentes sections.

**Le gradient**

**Le Laplacien** Le Laplacien peut s'écrire de la forme :

$$\Delta I(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

## 2.3 Méthode des différences finies

Nous cherchons à résoudre :

$$\begin{cases} \Delta I = \Delta S \ sur \ \Omega \\ I = T \ sur \ \partial\Omega \end{cases}$$

En utilisant la méthode des différences finie, commençons par discrétiser le Laplacien. Pour discrétiser celui-ci, il faut donc commencer par discrétiser $\frac{\partial^2 I}{\partial x^2}$ et $\frac{\partial^2 I}{\partial y^2}$.

**Discrétisation des dérivées secondes :** A l'aide des formules de Taylor à l'ordre 2 ci-dessous :

$$I(x+h,y) = I(x,y) + h \times \frac{\partial I(x,y)}{\partial x} + \frac{h^2}{2} \times \frac{\partial^2 I(x,y)}{\partial x^2} + o(h^3)$$
$$I(x-h,y) = I(x,y) - h \times \frac{\partial I(x,y)}{\partial x} + \frac{h^2}{2} \times \frac{\partial^2 I(x,y)}{\partial x^2} + o(h^3)$$

Et en effectuant la somme de ces deux équations, nous obtenons une discrétisation possible de $\frac{\partial^2 I(x,y)}{\partial x^2}$:

$$\frac{\partial^2 I(x,y)}{\partial x^2} = \frac{1}{h^2}\left(I(x+h,y) + I(x-h,y) - 2 \times I(x,y)\right)$$

### 2.3.1 Discrétisation du Laplacien

Comme le Laplacien n'est autre que la somme de $\frac{\partial^2 I(x,y)}{\partial x^2}$ et $\frac{\partial^2 I(x,y)}{\partial y^2}$, une discrétisation possible de celui-ci s'écrit de la forme :

$$\Delta I(x,y) = \frac{I(x+h,y) + I(x-h,y) - 2 \times I(x,y)}{h^2} + \frac{I(x,y+k) + I(x,y-k) - 2 \times I(x,y)}{k^2}$$

Les pas d'espaces h et k étant égaux à 1, nous pouvons écrire une discrétisation du Laplacien :

$$\Delta I(x,y) = I(x+1,y) + I(x-1,y) + I(x,y+1) + I(x,y-1) - 4 \times I(x,y)$$

**Application à une image** Afin de calculer le Laplacien du pixel I(i,j),il est donc nécessaire d'avoir la connaissance de ses pixels voisins que nous nommerons par la suite U(p), D(own), L(eft), R(ight) pour les pixels I(i-1,j), I(i+1,j), I(i,j-1), I(i,j+1).

| | | |
|---|---|---|
| | 1 | |
| 1 | -4 | 1 |
| | 1 | |

### 2.3.2 Résolution de l'équation de Poisson

Soit $g(x,y) = S(x+1,y) + S(x-1,y) + S(x,y+1) + S(x,y-1) - 4 \times S(x,y)$
Résoudre $\Delta I(x,y) = \Delta S(x,y)$ sur $\Omega$ est équivalent à résoudre :

$$\begin{cases} I(i+1,j) + I(i-1,j) + I(i,j+1) + I(i,j-1) - 4 \times I(i,j) = g(i,j) \\ \hfill pour (i,j) \in \Omega \\ \hfill I(i,j) = T(i,j) \ pour \ (i,j) \in \partial\Omega \end{cases}$$

Nous devons donc résoudre un système à $M \times N$ inconnues.

$$\begin{cases} \hfill I(1,1) = T(1,1) \\ \hfill I(3,2) + I(1,2) + I(2,3) + I(2,1) - 4I(2,2) = g(2,2) \\ \hfill I(3,3) + I(1,3) + I(2,4) + I(2,2) - 4I(2,3) = g(2,3) \\ \hfill ... \\ I(M,N-1) + I(M-2,N-1) + I(M-1,N) + I(M-1,N-2) = g(M-1,N-1) \\ \hfill I(M,N) = T(M,N) \end{cases} \tag{2}$$

Afin de résoudre ce système, il est plus facile de l'écrire sous forme matricielle. Nous devons donc résoudre un système de la forme AI = b et sa solution est : $I = A^{-1} \times b$. Avec :

- A une matrice carrée de taille ($M \times N$, $M \times N$)
- I un vecteur colonne de taille ($M \times N$,1)
- b un vecteur colonne de taille ($M \times N$,1)

Voici donc à quoi ressemble le système que nous souhaitons résoudre :

$$\begin{pmatrix} -4 & 1 & 0 & ... & 0 & 1 & 0... & 0 & ... & 0 \\ 1 & -4 & 1 & 0 & ... & 0 & 1 & 0 & .... & 0 \\ 0 & 1 & -4 & 1 & 0 & ... & 0 & 1 & 0 & ... \\ & ... & & & & & & & & \\ 1 & 0 & ... & 1 & -4 & 1 & 0... & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} I(1,1) \\ I(2,1) \\ ... \\ I(1,2) \\ I(2,2) \\ .... \\ I(M,N) \end{pmatrix} = \begin{pmatrix} g(1,1) \\ g(2,1) \\ ... \\ g(1,2) \\ g(2,2) \\ .... \\ g(M,N) \end{pmatrix} \tag{3}$$

**Transformation de l'image** Afin de pouvoir résoudre ce système, I doit être un vecteur colonne. Or dans notre cas, I est une matrice de taille (M,N). Le nouveau vecteur I, est la concaténation des colonnes de l'image I, comme ci dessous.

**Ecriture de la matrice du système** Pour remplir la matrice A dont nous avons besoin.Premièrement

**Ecriture de vecteur** Après avoir calculé le Laplacien de l'image S, le vecteur b est obtenu en concaténant les colonnes de l'image $\Delta S$.

**Exemple simple** Considérons l'image S ci-contre que nous souhaitons coller. Nous souhaitons ici coller les

Figure 7: Image à coller

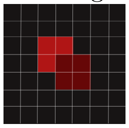


Figure 8: Vue grille pixel

deux carrés rouge sur une image que nous nommerons T. Commençons par sélectionner la zone que nous souhaitons coller, comme ci-dessous.

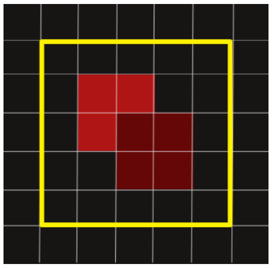


Figure 9: Sélection à coller

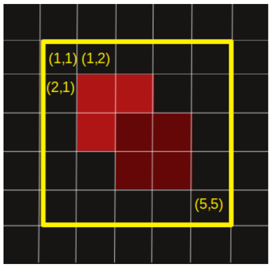



Figure 10: Numérotation des pixels

**Construction du système** Ici les pixels avec la pastille verte sont sur $\delta\Omega$. On veut donc résoudre le système suivant :

$$
\left\{
\begin{aligned}
I_{1,1} &= T_{1,1} \\
&... \\
I_{5,1} &= T_{5,1} \\
I_{1,2} &= T_{1,2} \\
\Delta I_{2,2} &= \Delta S_{2,2} \\
\Delta I_{3,2} &= \Delta S_{3,2} \\
\Delta I_{4,2} &= \Delta S_{4,2} \\
&...
\end{aligned}
\right. \tag{4}
$$

Avec $\Delta I(i,j) = U + D + L + R - 4I(i,j)$.
En écrivant ce système sous forme matricielle la matrice A est la matrice $25 \times 25$ ci-dessous :

$$
\left|
\begin{matrix}
1 & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & 1 & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & 1 & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & 1 & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & 1 & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & 1 & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & 1 & . & . & . & 1 & -4 & 1 & . & . & . & 1 & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & 1 & . & . & . & 1 & -4 & 1 & . & . & . & 1 & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & 1 & . & . & . & 1 & -4 & 1 & . & . & . & 1 & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & 1 & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & 1 & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & 1 & . & . & . & 1 & -4 & 1 & . & . & . & 1 & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & 1 & . & . & . & 1 & -4 & 1 & . & . & . & 1 & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & 1 & . & . & . & 1 & -4 & 1 & . & . & . & 1 & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & 1 & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & 1 & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & 1 & . & . & . & . & -4 & . & . & . & . & 1 & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & 1 & . & . & . & 1 & -4 & 1 & . & . & . & 1 & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & 1 & . & . & . & 1 & -4 & 1 & . & . & . & 1 & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & 1 & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & 1 & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & 1 & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & 1 & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & 1 & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & 1
\end{matrix}
\right.
$$

Figure 11: Matrice du système

Le vecteur I est obtenu en concaténant les colonnes de la sélection, il est de taille$(25 \times 1)$ :

$$
\begin{pmatrix}
I(1,1) \\
I(2,1) \\
... \\
I(5,1) \\
... \\
I(1,3) \\
... \\
I(5,3) \\
... \\
I(5,5)
\end{pmatrix} \tag{5}
$$

Enfin le vecteur b

$$
\begin{pmatrix}
T(1,1) \\
... \\
\Delta S(2,2) \\
... \\
T(5,2) \\
\Delta S(1,3)... \\
T(5,3) \\
\Delta S(2,4) \\
... \\
\Delta S(4,4) \\
T(5,4) \\
...
\end{pmatrix} \tag{6}
$$

La solution I, s'écrit sous la forme $I = A^{-1}b$.

## 2.4 Fourier

Avec cette seconde méthode nous allons résoudre l'équation de Poisson à l'aide de la transformée de Fourier. Avant de formuler la résolution de ce problème. Rappelons la définition des opérateurs dont nous aurons besoin dans la suite

### 2.4.1 Rappel et définitions des opérateurs

**Transformée de Fourier** Soit I une fonction, sa transformée de Fourier peut s'écrire de la façon suivante :

$$
\hat{I}(x,y) = \sum_{i=0}^{W-1} \sum_{j=0} H - 1 I(i,j) e^{-2\pi i \left( \frac{xi}{W} + \frac{yj}{H} \right)} \tag{7}
$$

**Gradient** Nous pouvons calculer le gradient d'une fonction dans le domaine de Fourier. Nous considérons toujours I la fonction que nous souhaitons étudier.

$$
\nabla(\hat{I}) = \begin{pmatrix} \frac{\partial \hat{I}}{\partial i} \\ \frac{\partial \hat{I}}{\partial j} \end{pmatrix} \tag{8}
$$

Avec

$$
\begin{aligned}
\frac{\partial \hat{I}}{\partial i} &= \left( \frac{2\pi i}{W} x \right) \hat{I} \\
\frac{\partial \hat{I}}{\partial j} &= \left( \frac{2\pi i}{H} y \right) \hat{I}
\end{aligned} \tag{9}
$$

Enfin afin de retrouver la fonction initiale nous aurons besoin de la transformée de Fourier inverse :

$$
I(i,j) = \frac{1}{WH} \sum_{x=0}^{W-1} \sum_{y=0}^{H-1} \hat{I}(x,y) e^{2\pi i \left( \frac{xi}{W} + \frac{yj}{H} \right)} \tag{10}
$$

### 2.4.2 Résolution de la méthode Fourier

# 3 Implémentation

Nous avons commencé à implémenter l'interface du projet. Voici une présentation de l'interface et ses fonctionnalités.

## 3.1 Interface

Voici comment se présente notre interface :

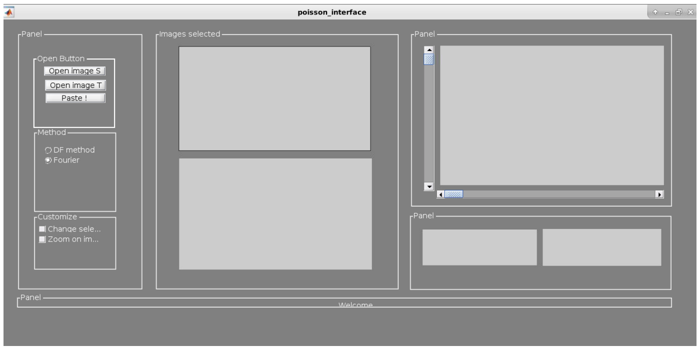



Figure 12: Interface 1.0

### 3.1.1 Ouverture d'images

Vous pouvez ouvrir n'importe quelle image en cliquant sur les boutons situés en haut à gauche de l'interface. EN cliquant sur eux, une boite de dialogue vous proposera de choisir une image présentes sur votre ordinateur. Une fois sélectionnée cette image sera affichée sur l'interface et nous pourrons travailler dessus. Image S correspond à l'image Source, c'est l'image que nous voulons coller. Et l'image T représente l'image Target, l'image sur laquelle nous souhaitons coller l'image S. C'est l'image d'arrière plan, en quelque sorte.
Ici nous avons choisi d'ouvrir ces deux images. L'image à coller s'affichera dans le premier rectangle, celui le plus haut. Tandis que l'image d'arrière plan s'affichera en dessous de la première.

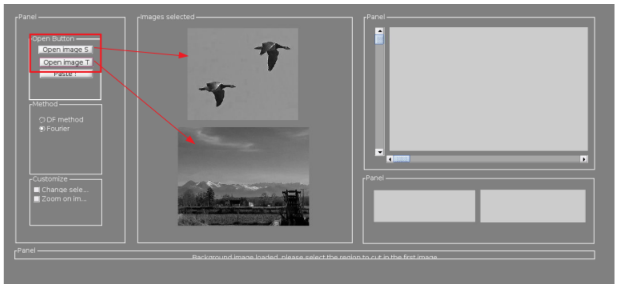



Figure 13: Ouverture d'images

### 3.1.2 Sélection

Une fois vos deux images ouvertes, il vous faut sélectionner la partie de l'image S que vous souhaitez coller dans l'image T. Pour cela cliquez une première fois sur l'image S : celle du haut puis cliquez une seconde fois pour dessiner sur l'image, la partie que vous souhaitez extraire.
Pour la seconde image, faites de même, cliquez une première fois sur l'image puis cliquez une seconde fois pour choisir l'endroit où vous souhaitez coller la partie sélectionnée plus tôt. Voici ce que vous devriez obtenir.

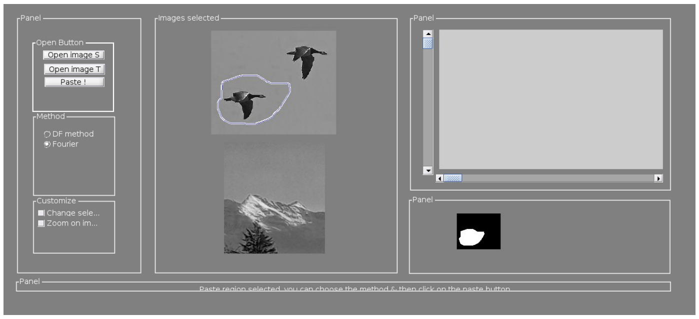



Figure 14: Sélection des parties

### 3.1.3 Choix des méthodes

Vous pouvez sélectionner la méthode avec laquelle vous voulez coller l'image S dans l'image T. Vous avez pour l'instant deux méthodes :

- Avec les différences finies
- Avec Fourier

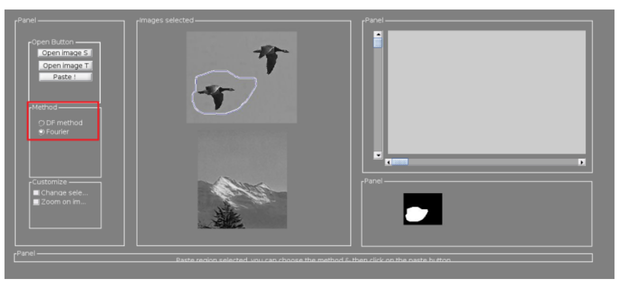



Figure 15: Choisir une méthode

### 3.1.4 Différences finies

Une fois la méthode DF choisie, il vous suffit de cliquer sur le bouton "Paste!" pour afficher le résultat. Ce résultat s'affiche dans la partie droite de l'interface.


Figure 16: Différences finies

#### 3.1.5 Fourier

Pour afficher le résultat obtenus avec Fourier, il suffit de cocher la case Fourier  Nous avons ajoutés des options


Figure 17: Résultat obtenu avec la méthode de Fourier

comme l'option de zoom qui permet de zoomer sur une image pour voir de plus près le collage de l'image.


Figure 18: Zoom

#### 3.1.6 Amélioration avec change selection

**ATTENTION CETTE FONCTIONNALITE NE FONCTIONNE POUR L'INSTANT QU'AVEC LA METHODE DES DIFFERENCES FINIES** Nous avons aussi remarquer un problème en effectuant différents tests sur les images :

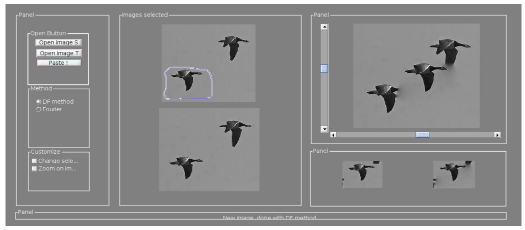


Figure 19: Problème de collage

Ainsi nous avons ajouté, l'option Change sélection qui re-dimensionne automatique la sélection afin de résoudre ce pb.

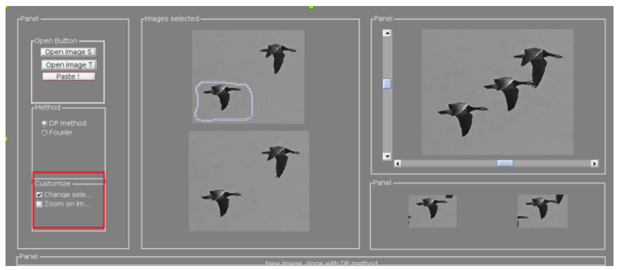



Figure 20: Résolution du problème

### 3.1.7 Sliders

Vous pouvez bouger les sliders présents à droite afin de déplacer la zone collée dans l'image de fond. La solution au problème est alors automatiquement recalculée en fonction de l'endroit où l'objet à collé.

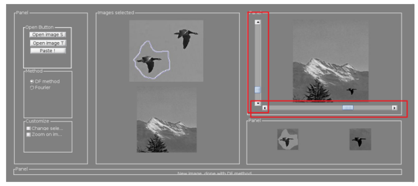


Figure 21: Modifier l'endroit de collage

# 4 Résultats obtenus

# 5 Comparaison

# 6 Optimisation

## 6.1 Méthode de Douglas

# 7 Conclusion