

Inpainting

Livio

Juin 2024

Contents

1	Introduction	2
2	Fast digital Inpainting	2
2.1	Capacités de notre Méthode d’Inpainting	2
2.2	Limitations de notre Méthode d’Inpainting	2
2.3	Prise en compte des bords	2
2.4	Résultats	2
2.5	Limite liée aux Bords	3

1 Introduction

2 Fast digital Inpainting

Dans le contexte de notre algorithme d'inpainting simple, il est vrai que nous ne pouvons pas recréer des détails complexes ou du contenu contextuel sophistiqué dans les zones manquantes. Notre approche repose principalement sur une diffusion isotrope simple, qui est essentiellement un lissage des valeurs de pixels voisins. Voici les limitations et les capacités de cette méthode :

2.1 Capacités de notre Méthode d'Inpainting

Lissage et Propagation Simples:

La méthode de diffusion simple que nous utilisons peut lisser les bords et propager les valeurs de pixels voisins dans les zones masquées. Elle peut être efficace pour combler de petites imperfections ou des zones où les informations de pixels sont relativement homogènes.

Restitution des Petites Zones:

Pour de petites zones manquantes, l'algorithme peut fournir une approximation visuellement acceptable en diffusant les valeurs des pixels environnants. Idéal pour supprimer des petites imperfections, des poussières, des rayures fines, etc.

2.2 Limitations de notre Méthode d'Inpainting

Manque de Détails Complexes:

L'algorithme de diffusion simple ne peut pas recréer des motifs ou des structures complexes. Par exemple, il ne peut pas redessiner une texture complexe ou un motif précis comme un visage humain ou des détails architecturaux fins. Il ne peut pas comprendre ou recréer le contexte sémantique d'une image, ce qui signifie qu'il ne peut pas deviner les objets ou les structures qui devraient être dans les zones manquantes.

Floutage

La diffusion isotrope a tendance à introduire du flou, en particulier lorsque des zones de contraste élevé sont impliquées. Cela signifie que les bords et les détails fins peuvent devenir flous après l'inpainting.

2.3 Prise en compte des bords

En ajoutant les contours détectés à l'algorithme, on crée des barrières de diffusion qui empêchent le processus d'inpainting de modifier les pixels le long des bords détectés. Cela garantit que les bords et les contours de l'image d'origine sont préservés autant que possible dans l'image inpainted finale, ce qui donne un résultat visuellement plus fidèle à l'image originale.

2.4 Résultats

Pour tester notre cas nous aller créer un masque manuellement. C'est un carré de 6x6. Avec 30 itérations et un noyau gaussien, voici un type de résultat que nous pouvons obtenir :

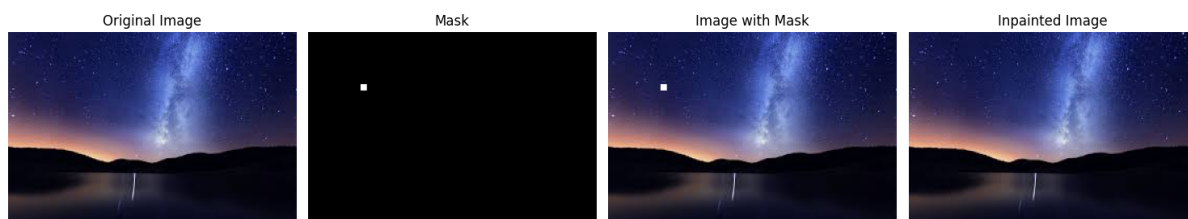


Figure 1: Exemple de fast digital inpainting

2.5 Limite liée aux Bords

Nous avons vu que notre algo fonctionne correctement. Cependant, ce dernier ne conserve pas la structure de l'image. Ainsi si il est placé sur un edge de l'image, la reconstruction sera décevante.

Pour mettre en valeur cela, on calcule les edges avec l'algorithme de Canny. Il est l'un des algorithmes les plus largement utilisés pour la détection de contours dans les images.

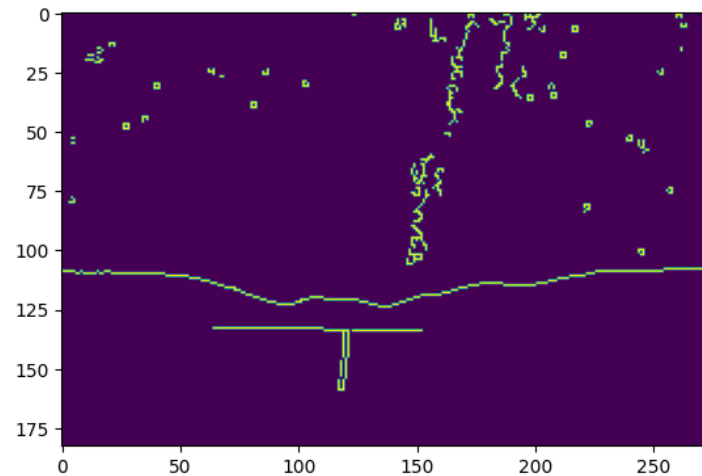


Figure 2: Bords de l'image précédente

Maintenant plaçons notre carré blanc sur l'un des bords détectés et observons le résultat.

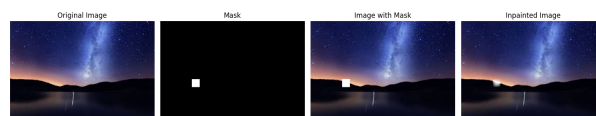


Figure 3: Enter Caption

On observe que même après un grand nombre d'itération (ici Niter = 500), le lissage ne s'effectue pas.

Il reste encore à tester l'influence de la taille du noyau et le type de noyau choisi. Ici on n'a utilisé qu'un noyau gaussien.