

Object Removal by Exemplar-Based Inpainting

Clément Gilli
Jean Groëninger

Novembre 2024

Table des matières

- 1 Introduction
- 2 Formalisation du problème
- 3 Implémentation et choix faits
- 4 Utilisation
- 5 Résultats
- 6 Améliorations possibles
- 7 Conclusion

Introduction

- Algorithme simple :
 - ① Choisit un patch à remplir sur la frontière entre la zone connue et la zone à remplir
 - ② Cherche un patch dans toute l'image qui se rapproche le plus du patch choisi
 - ③ Colle le patch trouvé au bon endroit
- Spécificité de [1]

Table des matières

- 1 Introduction
- 2 Formalisation du problème
- 3 Implémentation et choix faits
- 4 Utilisation
- 5 Résultats
- 6 Améliorations possibles
- 7 Conclusion

Notation

- Φ : la région d'origine
- Ω : la région masquée à compléter
- $\delta\Omega$: la frontière de la région masquée
- Ψ_p : un patch de taille $k \times k$ pixels, centré en p

L'algorithme

① Calcul des priorités

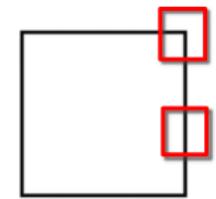
- Priorité $P(\mathbf{p}) = C(\mathbf{p})D(\mathbf{p})$

$$C(\mathbf{p}) = \frac{\sum_{\mathbf{q} \in \Psi_{\mathbf{p}} \cap \bar{\Omega}} C(\mathbf{q})}{|\Psi_{\mathbf{p}}|}$$

$$D(\mathbf{p}) = \frac{|\nabla I_{\mathbf{p}}^{\perp} \cdot \mathbf{n}_{\mathbf{p}}|}{\alpha}$$

② Propagation des structures

- $\Psi_{\hat{\mathbf{p}}} | \hat{\mathbf{p}} = \arg \max_{\mathbf{p} \in \delta\Omega} P(\mathbf{p})$
- $\Psi_{\hat{\mathbf{q}}} = \arg \min_{\Psi_{\mathbf{q}} \in \Phi} d(\Psi_{\hat{\mathbf{p}}}, \Psi_{\mathbf{q}})$



③ Mise à jour des termes de confidence : $\forall \mathbf{q} \in \Psi_{\hat{\mathbf{p}}} \cap \Omega, C(\mathbf{q}) = C(\hat{\mathbf{p}})$

Table des matières

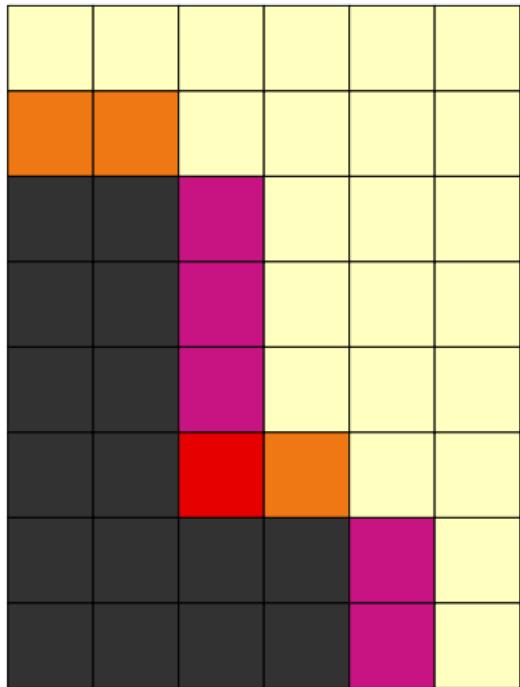
- 1 Introduction
- 2 Formalisation du problème
- 3 Implémentation et choix faits
- 4 Utilisation
- 5 Résultats
- 6 Améliorations possibles
- 7 Conclusion

Structure de données



Figure: Tailles de patch (gauche 9x9, droite 21x21)

Calcul de ∇I_p sur la frontière



Calcul du gradient possible



Calcul de la première composante
du gradient possible



Calcul de la seconde composante
du gradient possible



Calcul du gradient impossible

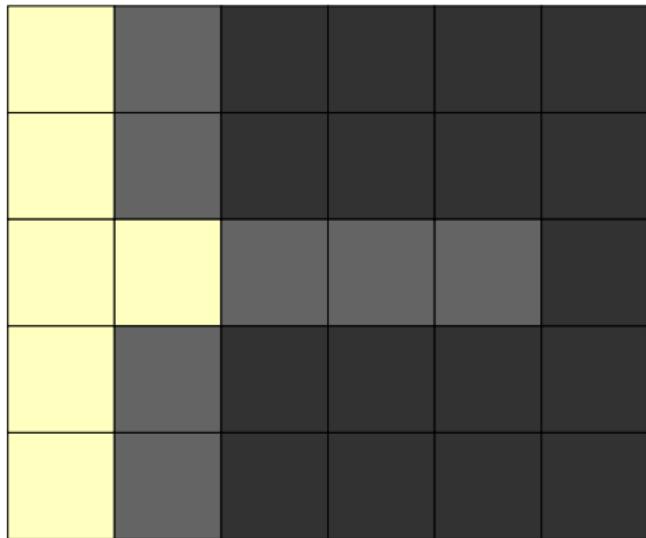


Zone cible

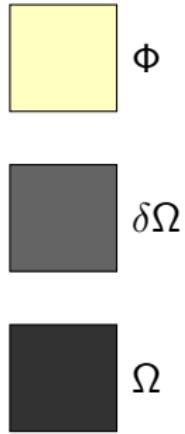
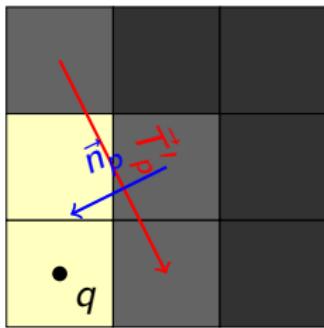
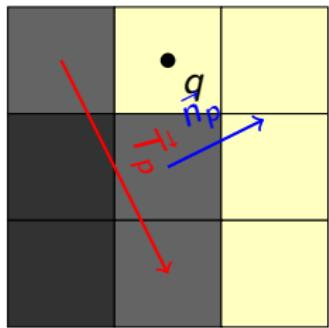
Calcul de ∇I_p sur la frontière

$$\nabla I_{(a,b)} = \begin{cases} \nabla I_{(a,b)}, & \text{si } (a, b) \in \Phi, \\ \left(\frac{\partial}{\partial a} I(a, b), \frac{\partial}{\partial b} I(a, b + \varepsilon)\right)^T, & \text{si } (a, b) \in \delta\Omega \text{ et } \exists \varepsilon \in \{-1, 1\}, (a, b + \varepsilon) \in \Phi \\ \left(\frac{\partial}{\partial a} I(a + \varepsilon, b), \frac{\partial}{\partial b} I(a, b)\right)^T, & \text{si } (a, b) \in \delta\Omega \text{ et } \exists \varepsilon \in \{-1, 1\}, (a + \varepsilon, b) \in \Phi \\ \nabla I_{(a+\varepsilon_1, b+\varepsilon_2)} & \text{si } (a, b) \in \delta\Omega \text{ et} \\ & \exists \varepsilon_1, \varepsilon_2 \in \{-1, 1\}, (a + \varepsilon_1, b + \varepsilon_2) \in \Phi \\ 0 & \text{sinon} \end{cases}$$

Calcul de ∇I_p sur la frontière

 Φ  $\delta\Omega$  Ω

Calcul de n_p



Recherche du plus proche voisin

Method	Construction Performance	ϵ -NN Search Performance	k -NN Search Performance
kd -Tree	Excellent	Poor	Poor
PCA Tree	Poor	Fair	Fair
Ball Tree	Fair	Excellent	Excellent
k -Means	Poor	Good	Good
vp -Tree	Excellent	Excellent	Excellent

Figure: Tableau récapitulatif des méthodes de recherche [2]

Recherche du plus proche voisin

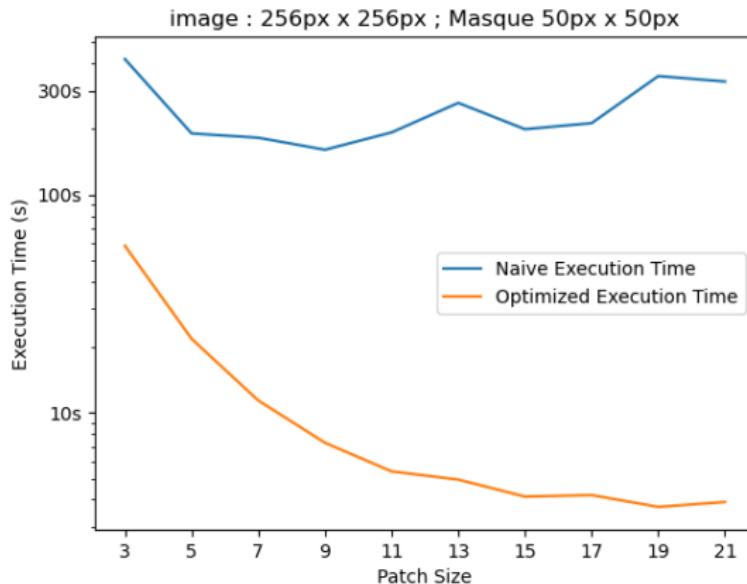


Figure: Temps d'exécution des deux méthodes en fonction de la taille des patchs

Zone de recherche

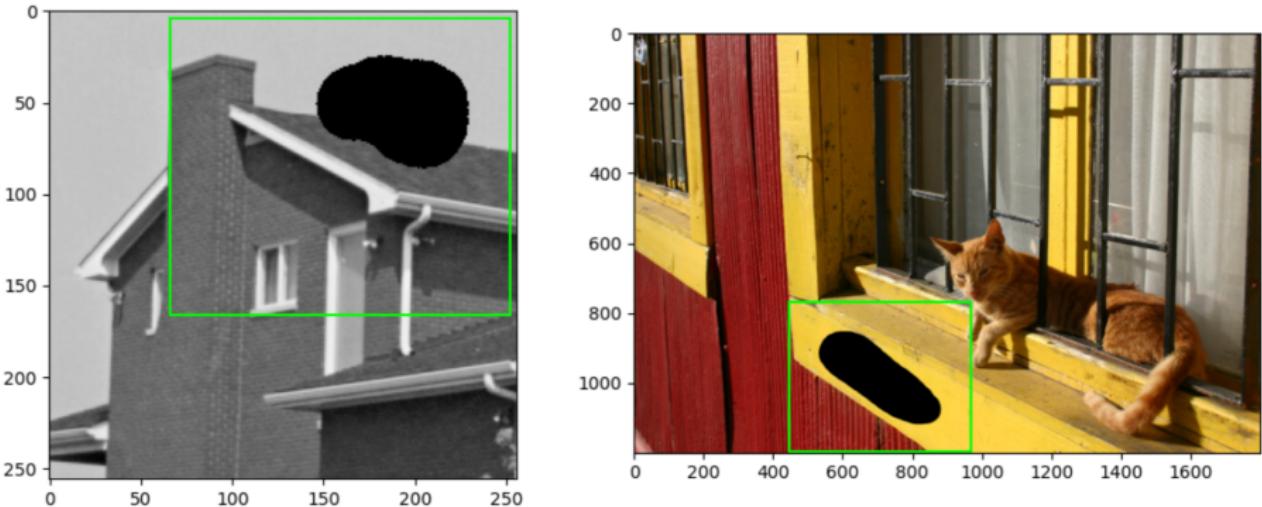


Figure: En vert la zone de recherche

Zone de recherche

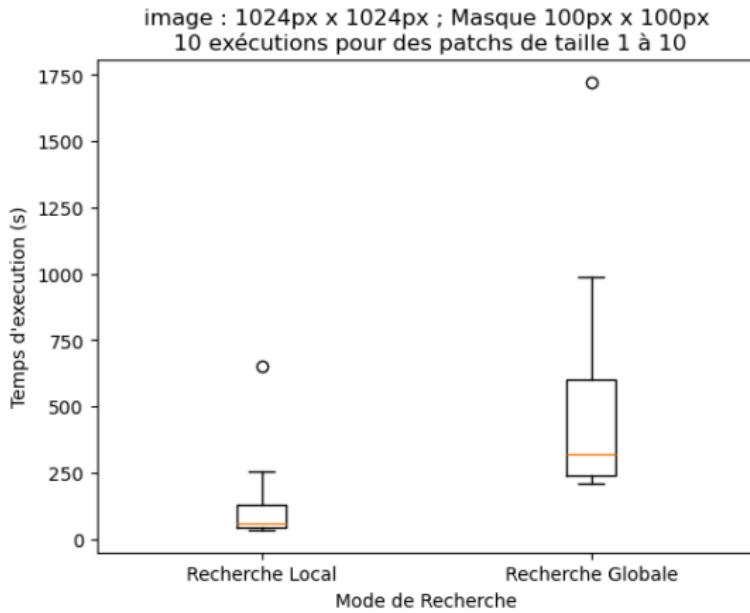


Figure: BoxPlot du temps d'exécution pour la Recherche Locale et Globale

Table des matières

- 1 Introduction
- 2 Formalisation du problème
- 3 Implémentation et choix faits
- 4 Utilisation
- 5 Résultats
- 6 Améliorations possibles
- 7 Conclusion

Table des matières

- 1 Introduction
- 2 Formalisation du problème
- 3 Implémentation et choix faits
- 4 Utilisation
- 5 Résultats
- 6 Améliorations possibles
- 7 Conclusion

Importance de la taille des patchs



Masque en rouge

Importance de la taille des patchs



9x9



13x13



17x17



21x21

Résultats



Résultats



Résultats



Résultats



Figure: Propagation des structures parfaite

Cas des textures

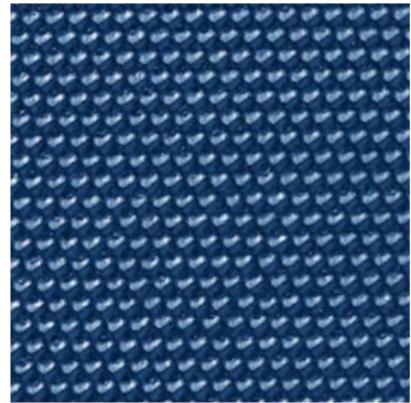
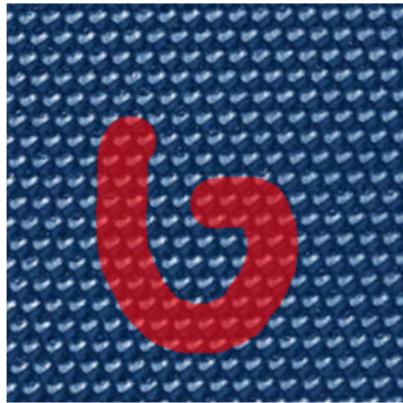
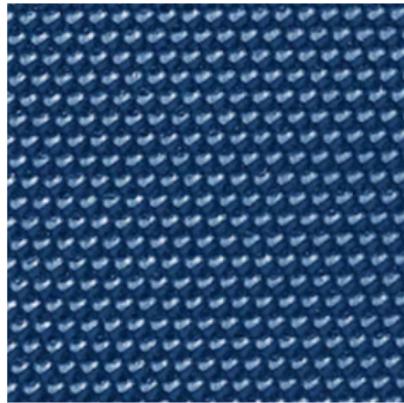


Figure: Motif répété

Cas des textures

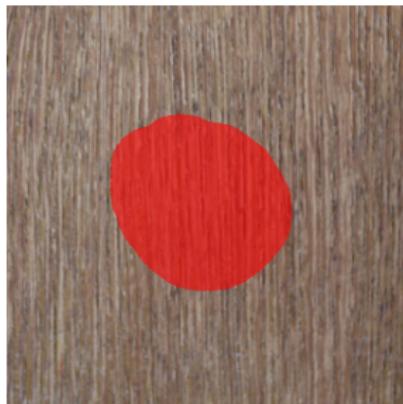


Figure: Gaussien

Cas des textures

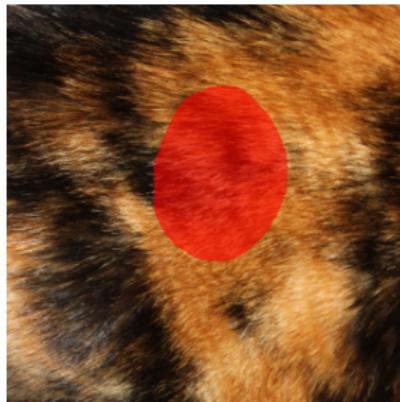


Figure: Non-gaussian

Table des matières

- 1 Introduction
- 2 Formalisation du problème
- 3 Implémentation et choix faits
- 4 Utilisation
- 5 Résultats
- 6 Améliorations possibles
- 7 Conclusion

Rendu visuel

1ère idée [3]

$$\hat{P}(\mathbf{p}) = C(\mathbf{p}) \exp\left(-\frac{D(\mathbf{p})}{2\sigma^2}\right)$$

2ème idée

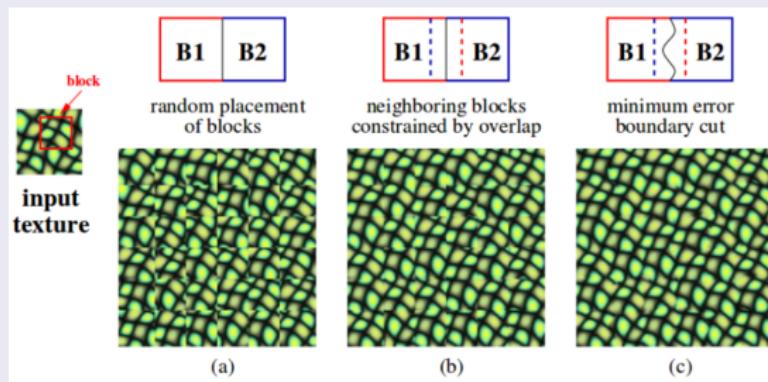


Figure: Méthodes de recollement (Efros-Freeman 2001)

Complexité

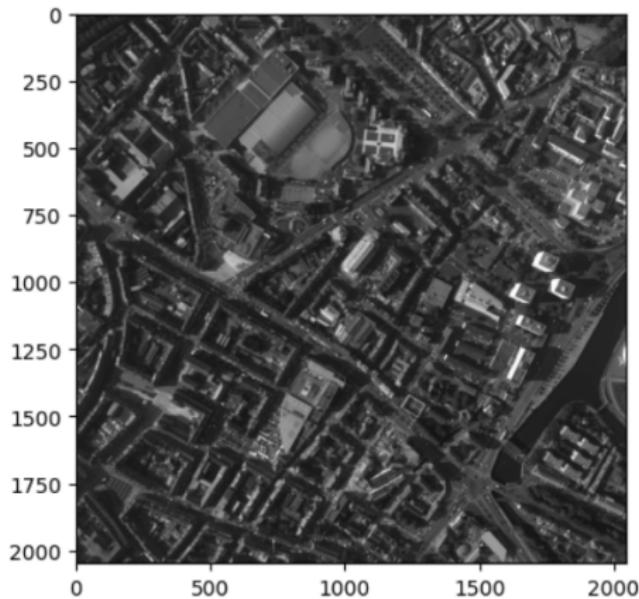
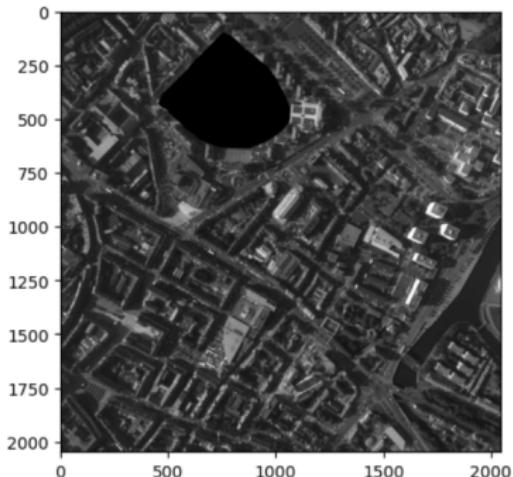
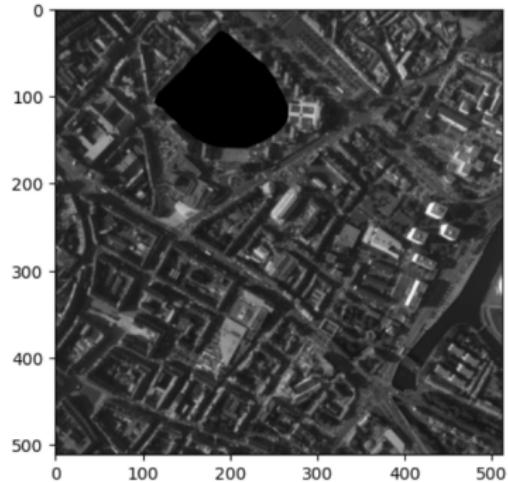


Image originale

Complexité

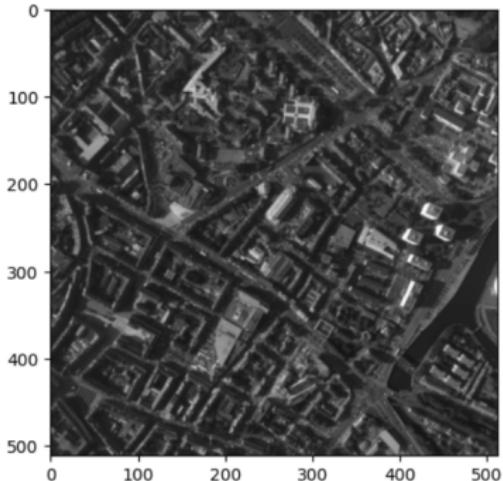


Avec masque

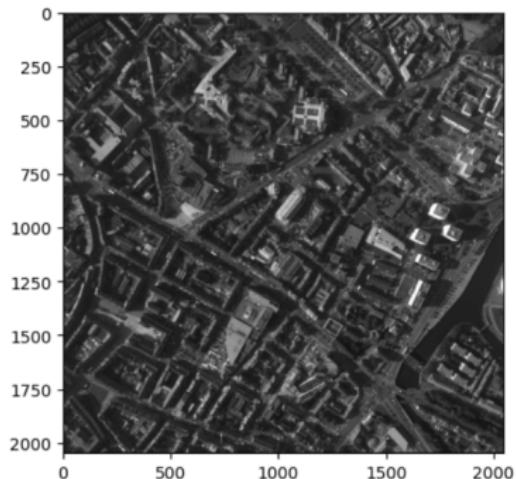


Sous-échantillonnée

Complexité



Reconstruction



Retour taille originale

Table des matières

- 1 Introduction
- 2 Formalisation du problème
- 3 Implémentation et choix faits
- 4 Utilisation
- 5 Résultats
- 6 Améliorations possibles
- 7 Conclusion

Bibliographie

- [1] A. Criminisi, P. Pérez, and K. Toyama. "Object Removal by Exemplar-Based Inpainting". In: *Microsoft Research*. 2003.
- [2] Neeraj Kumar, Li Zhang, and Shree Nayar. "What is a Good Nearest Neighbors Algorithm for Finding Similar Patches in Images?" In: *Columbia University*. 2008.
- [3] Raul Martinez Noriega, Aline Roumy, and Gilles Blanchard. "Exemplar-based image inpainting: fast priority and coherent Nearest Neighbor search". In: *IEEE Workshop on Machine Learning for Signal Processing*. 2012.