



École des Ponts

ParisTech

École des Ponts ParisTech
2022 - 2023

Rapport de Projet

Philomène BOISNARD

Erwann ESTEVE

Wandrille FLAMANT

Sixtine NODET

Élèves ingénieurs

Projet de Semestre 2

TRUCAGE PHOTOGRAPHIQUE PAR INPAINTING

Février - Juin 2023

Encadrant :
Pascal MONASSE

Table des matières

Introduction	2
1 Présentation de l'algorithme	3
1.1 Les objets exploités	3
1.2 L'initialisation	3
1.2.1 Initialisation des pixels hors de la zone Ω	3
1.2.2 Initialisation de la frontière $\partial\Omega$	3
1.3 Calcul des termes <i>Priorité</i> , <i>Confiance</i> et <i>Data</i>	3
1.3.1 Calcul des priorités $P(p)$	3
1.3.2 Calcul de la Confiance (ou <i>Confidence</i>) $C(p)$	3
1.3.3 Calcul du terme <i>Data</i> $D(p)$	4
1.4 Le <i>Matching</i> et le calcul des distances	4
1.5 Mise à jour de la <i>target zone</i> Ω et de la frontière $\partial\Omega$	4
2 Analyse des résultats	5
2.1 Sous-partie	5
2.2 Sous-partie	5
2.3 Sous-Partie	5
3 Critique de l'algorithme	6
3.1 La complexité	6
3.2 Les cas limites	6
3.3 Pistes d'amélioration	6
Conclusion	7
Références	8
Annexes	9
Captures d'écran	9

Introduction

Notre projet se base principalement sur le papier de A.Criminisi [1].

Dans un premier temps, notre travail s'est concentré sur la compréhension du document et de l'algorithme expliqué.

L'**inpainting** est le nom donné à la technique de reconstruction d'images détériorées ou de remplissage des parties manquantes d'une image

L'**objectif principal** de notre projet va être de construire un algorithme capable d'effectuer ce trucage par inpainting, idéalement dans une complexité spatiale et temporelle maîtrisée, puis d'en trouver les limites.

1 Présentation de l'algorithme

De la lecture du document [1], nous en avons déduit l'**algorithme 1** suivant en pseudo-code qui permet de résoudre le problème du trucage par inpainting.

Algorithm 1 Algorithme de remplissage de la zone target

Require: Image, target zone Ω

Initialiser les pixels hors de Ω

Initialiser de la frontière $\partial\Omega$

while $\partial\Omega \neq \emptyset$ **do**

 Calculer les priorités $P(p)$, $\forall p \in \partial\Omega$

 Trouver le pixel $p \in \partial\Omega$ maximisant $P(p)$

 Trouver le patch Ψ_q minimisant la distance avec Ψ_p

 Copier Ψ_q sur $\Psi_p \cap \Omega$

 Calculer $C(p)$, $\forall p \in \Psi_p \cap \Omega$

 Mettre à jour Ω et $\partial\Omega$

end while

Nous avons fragmenté le travail en plusieurs sous problèmes que nous avons ensuite réparti entre les différents membres du groupe.

Les sous problèmes sont notamment ceux du traitement de la frontière de la *target zone*, de la gestion des paramètres **confidence** et **data**, l'affichage des images, le matching d'un *patch*, etc.

1.1 Les objets exploités

La *target zone* Ω , un patch Ψ_p , la frontière $\partial\Omega$.

Pour mieux comprendre les objets appartenant aux classes **Pixel**, **Image** et **Frontiere**, les choix d'implémentation sont développés en Annexe.

1.2 L'initialisation

1.2.1 Initialisation des pixels hors de la zone Ω

Texte

1.2.2 Initialisation de la frontière $\partial\Omega$

Texte

1.3 Calcul des termes *Priorité*, *Confiance* et *Data*

1.3.1 Calcul des priorités $P(p)$

Texte

1.3.2 Calcul de la Confiance (ou *Confidence*) $C(p)$

Texte

1.3.3 Calcul du terme *Data* $D(p)$

Texte

1.4 Le *Matching* et le calcul des distances

Texte

1.5 Mise à jour de la *target zone* Ω et de la frontière $\partial\Omega$

Texte

2 Analyse des résultats

2.1 Sous-partie

Texte

2.2 Sous-partie

Texte

2.3 Sous-Partie

Texte

3 Critique de l'algorithme

3.1 La complexité

Spatiale et temporelle

Texte

3.2 Les cas limites

Texte

3.3 Pistes d'amélioration

Texte

Conclusion

Retour d'expérience.

Delta.

Améliorations.

Références

- [1] Antonio CRIMINISI, Patrick PÉREZ et Kentaro TOYAMA. « Region filling and object removal by exemplar-based image inpainting ». In : *IEEE Transactions on image processing* 13.9 (2004), p. 1200-1212.

Annexes

Les classes principales

La classe Pixel

Texte

La classe Image

Texte

La classe Frontière

Texte

Captures d'écran

Les bo screens waa