Inpainting vidéo préservant le mouvement

Thuc Trinh LE[1,2], Andrés ALMANSA[1], Yann GOUSSEAU[2], Simon MASNOU[3]

[1]MAP5, CNRS & Université Paris Descartes,
75006 Paris, France

[2]LTCI, Télécom ParisTech, Université Paris-Saclay,
46 Rue Barrault, 75013 Paris, France

[3]Univ Lyon, Univ Claude Bernard Lyon 1 & CNRS, Institut Camille Jordan,
69622 Villeurbanne, France

thuc.le@enst.fr, andres.almansa@enst.fr
yann.gousseau@enst.fr, masnou@math.univ-lyon1.fr

Résumé – Dans cet article, nous proposons une technique rapide et automatique pour l'inpainting vidéo permettant de traiter des scènes réelles en haute définition, notamment en présence de mouvements arbitraires tels les mouvements de caméra, les mouvements rapides et prolongés ou les mouvements complexes. Notre approche repose sur l'optimisation d'une énergie globale fondée sur la cohérence des patches de la vidéo. Pour maintenir la cohérence temporelle, le flot optique est utilisé systématiquement aux différentes étapes de l'algorithme. Des expériences sur de nombreuses vidéos montrent que notre technique présente des avantages conséquents en comparaison de travaux antérieurs, à la fois en termes de précision et de temps de calcul.

Abstract – In this paper, we propose a fast and automatic inpainting technique for real video data which works under many challenging conditions such as moving camera, dynamic scenes or complex motion. The video inpainting problem is solved by optimizing a global patch-based function. To maintain the temporal consistency, the optical flow is used systematically in the various stages of the algorithm. Experiments on both classical and new challenging sequences show that our technique outperforms state-of-the-art methods in terms of accuracy and computational time.

1 Introduction et état de l'art

L'inpainting vidéo consiste à réinventer le contenu d'une zone spatio-temporelle manquante dans une vidéo de sorte que le résultat obtenu soit visuellement plausible. Une façon classique de procéder consiste à utiliser l'information spatiale et temporelle des zones voisines. L'inpainting vidéo a de nombreuses applications telles que l'édition vidéo, la super-résolution spatiale et temporelle, ou la restauration de films anciens. Il s'agit d'un problème relativement peu exploré en raison de sa difficulté – en particulier pour garantir la cohérence temporelle – et de la complexité des calculs qu'il nécessite. Plusieurs méthodes aux résultats prometteurs ont toutefois été proposées ces dernières années. La plupart d'entre elles utilisent principalement deux approches : celles qui impliquent une détection des éléments constitutifs de la scène, que l'on appelle "orientées objets", et les approches reposant sur les patches et tirant partie des auto-similarités des scènes.

Dans les approches orientées objets, une étape de prétraitement est nécessaire pour décomposer la vidéo en séparant le fond des objets présentant un mouvement significatif. Cette approche donne généralement des résultats raisonnables lorsqu'il faut traiter un objet spécifique. Cependant, les objets en mouvement et le fond étant restaurés indépendamment, la recomposition peut générer des artefacts.

Dans les approches reposant sur les patches, la partie à restaurer est reconstruite au moyen des patches de la partie connue, ou partie source. On peut par exemple mentionner les approches "gloutonnes", fortement séquentielles, dans lesquelles les pixels sont restaurés les uns après les autres, en utilisant pour chacun d'entre eux le ou les "meilleurs" patches dans l'image source. L'ordre de reconstruction est régi par un terme de priorité, dont l'influence sur le résultat final est fortement non linéaire, ce qui limite la robustesse de ce type de méthodes, en particulier lorsque les domaines à reconstruire sont de grande taille. De plus, ces approches gloutonnes ne permettent généralement pas de restaurer des structures ou des mouvements sur de grandes distances ou de grands intervalles de temps.

Pour garantir la cohérence, géométrique comme de mouvement, une approche globale est nécessaire. Le principe le plus efficace – à notre connaissance – est de minimiser une fonction globale reposant sur la cohérence de patches. Cette méthode, initialement proposée dans Wexler et al. [1], a fait l'objet de plusieurs améliorations. Newson et al. [2] ont notamment proposé l'utilisation d'une variante 3D de l'algorithme PatchMatch [3] pour renforcer la cohérence et accélérer la vitesse de l'algorithme. Granados et al. [4] se sont concentrés sur l'estimation de la carte des plus proches voisins dans l'espace 3D

en utilisant des coupes de graphe. Récemment, Huang et al. [5] ont modifié l'énergie de Wexler et al.[1] en y ajoutant un terme de flot optique pour imposer la cohérence temporelle, mais en utilisant des patches 2D et non 3D.

Cependant, ces méthodes présentent certains inconvénients tels qu'un temps de calcul très conséquent [4], une incapacité à traiter le mouvement sur une période longue [2] ou des artefacts indésirables [5]. Pour résoudre ces problèmes, nous proposons une technique de complétion de vidéo basée sur les méthodes de Wexler et al. [1] et Newson et al. [2] avec trois nouvelles contributions. La première consiste à intégrer fortement le flot optique. Il est utilisé pour définir une nouvelle distance entre patches, pour guider la recherche de patches similaires, permettre une séparation implicite des objets et de l'arrière-plan et calculer un terme de priorité dans la phase cruciale d'initialisation de l'algorithme. Cette prise en compte systématique de l'information de mouvement nous permet d'assurer la cohérence temporelle et la reconstruction d'objets mobiles dans une occultation longue. La deuxième contribution est une réduction significative du temp de calcul, obtenue en parallélisant l'algorithme PatchMatch 3D. La dernière contribution est l'intégration d'une carte de confiance et d'une carte de séparation dans l'étape finale de reconstruction des pixels à partir des patches plus proches voisins. Nous évaluons notre méthode sur des vidéos très diverses et comparons les résultats à des approches récentes. Ces résultats sont disponibles sous forme de complément à cet article.

2 Méthode proposée

2.1 Présentation générale

Notre méthode repose sur une approche non-locale basée sur les patches à l'instar de [1] et [2]. Une énergie non-convexe est optimisée par une procédure itérative intégrée dans un schéma pyramidal *coarse-to-fine*. L'optimisation comporte deux étapes fondamentales : une étape de recherche de patches qui estime le champ des plus proches voisins et une étape de reconstruction utilisant ce champ pour déterminer les valeurs de chaque pixel dans la zone à reconstruire.

Dans un tel cadre, et comme souvent pour les problèmes d'inpainting, le diable se cache dans les détails. De nombreux problèmes doivent être pris en considération, tels que la déformation des patches causée par le mouvement de la caméra, la complexité algorithmique, l'étape d'initialisation, la stratégie de recherche pour trouver des patches appropriés et la préservation du mouvement. Ces problèmes sont résolus dans notre méthode en utilisant une version parallèle de PatchMatch, en proposant une stratégie de recherche et d'appariement de patches exploitant le flot optique et en intégrant un nouveau schéma d'initialisation. Ces techniques seront présentées dans les sections suivantes.

2.2 Energie

Pour gérer l'instabilité provoquée par les mouvements de caméra, un prétraitement de stabilisation est effectué. Après la stabilisation, nous minimisons une énergie $E(u, \phi)$ pour trouver la séquence reconstruite u et l'application ϕ de transfert de patches. En désignant par W_p^u le patch centré sur un pixel p dans la région inconnue $\mathcal{H}$, le transfert $\phi(p)$ est défini comme le décalage spatial $q - p$ où q est un minimiseur dans $\mathcal{H}^c$ de la distance $d(W_p^u, W_q^u)$ (voir ci-dessous). L'énergie E associée à une image u et une carte de transfert ϕ est alors définie par :

$$E\left(u,\phi\right)=\sum_{p\in\mathcal{H}}d^2\left(W_p^u,W_{p+\phi(p)}^u\right).$$

La métrique d entre patches est définie par

$$d^2\left(W_p^u,W_q^u\right)=\frac{1}{|N_p|}\sum_{\substack{r\in N_p\\ r-p+q\notin\mathcal{H}}}[\alpha\left(\|u(r)-u(r-p+q)\|_2^2\right)+$$

$$\beta\left(\|T(r)-T(r-p+q)\|_2^2\right)+\gamma\left(\|O(r)-O(r-p+q)\|_2^2\right)],$$

où N_p indique le domaine spatio-temporel du patch centré sur p. Au lieu d'un pavé droit, nous utilisons un parallélépipède dont la forme est contrôlée par le flot optique de façon à mieux capturer l'information de mouvement. Le vecteur T a pour composantes $(\left|\frac{\partial u}{\partial x}\right|,\left|\frac{\partial u}{\partial y}\right|)$ et permet de décrire les caractéristiques locales de texture, le vecteur $O = (|O_x|, |O_y|)$ est constitué des modules des deux composantes du vecteur de flot optique et α, β, γ sont des coefficients de pondération. Ces descripteurs ont vocation à éviter les mises en correspondance erronées, qui sont potentiellement fréquentes en raison d'oscillations spatiales ou temporelles dans les vidéos.

L'énergie E est non-convexe et de dimension relativement élevée. Néanmoins, on obtient en pratique des résultats raisonnables par minimisation alternée par rapport à u et ϕ, combinée avec une bonne initialisation et à un schéma multi-échelles. Ce schéma multi-échelles consiste à effectuer l'inpainting sur les niveaux successifs d'une pyramide gaussienne, comme proposé dans [1] et [2], mais en utilisant trois pyramides, correspondant aux informations de couleur, texture et flot optique utilisées pour comparer les patchs. La structure générale de notre algorithme est la suivante :

- Construction des pyramides pour la couleur u, le domaine d'occlusion $\mathcal{H}$, les caractéristiques de texture T et de mouvement O.
- Initialisation à l'échelle la plus grossière (voir la section 2.3).
- De l'échelle la plus grossière à l'échelle la plus fine :
  - Itérer jusqu'à convergence :
    - $\min_\phi$ (Recherche des plus proches voisins, section 2.4).
    - $\min_u$ (Reconstruction du pixel, section 2.5).
    - Reconstruction des caractéristiques.
  - Si l'on n'est pas à l'échelle la plus fine : interpoler ϕ, reconstruire u et les caractéristiques T, O.

2.3 Initialisation grossière

En raison de la non-convexité de l'énergie, une initialisation fiable est nécessaire pour avoir une bonne minimisation locale. Cette étape cruciale est souvent passée sous silence dans la littérature. Récemment, Newson et al. [2] ont proposé une technique de reconstruction graduelle dans laquelle l'ordre des régions à reconstruire est donné par un parcours en spirale, c'est à dire couche par couche de la frontière de la zone à reconstruire vers le centre. Ce schéma fonctionne bien tant que la zone à reconstruire n'est pas trop étendue. Cependant, dans le cas où les objets en mouvement sont perdus pendant une longue période, ce schéma a tendance à les faire disparaître, et ceux-ci ne peuvent plus être reconstruits.

Pour pallier ce problème, nous proposons une initialisation exploitant l'information de flot optique pour imposer une priorité dans l'ordre de reconstruction. Plus précisément, le terme de priorité du pixel i est défini comme $Pr_i = C_i.D_i$, où D_i est la moyenne de l'amplitude du flot optique dans un patch centré sur le pixel i et $C_i \propto \exp(-d^2(i, H_{coarse}))$ mesure la distance du pixel i à la frontière de la region à reconstruire à l'échelle la plus grossière, notée H_{coarse}. Après seuillage du flot optique par la méthode d'Otsu [6] (sans paramètres), afin d'effectuer une distinction entre pixels présentant un mouvement significatif et pixels de fond, on répète cette procédure jusqu'à ce que tous les pixels des objets en mouvement aient été reconstruits :

— On définit le bord tubulaire de la région occultée $\mathcal{H}'$ comme $\mathcal{B}' = \mathcal{H}' \setminus (\mathcal{H}' \ominus B(0,1))$. On calcule Pr_i pour $i \in \mathcal{B}'$.
— On sélectionne le patch P_i qui a la plus grande priorité Pr_i, et on définit la région à reconstruire $R_i = P_i \cap \mathcal{B}'$.
— On reconstruit R_i (par le plus proche voisin de P_i) et on définit la nouvelle région occultée $\mathcal{H}' \rightarrow \mathcal{H}' \setminus R_i$.

Enfin, ce qui reste de la région à reconstruire (le fond), est restauré par la méthode de Newson et al. [2].

2.4 Estimation des plus proches voisins

Dans notre méthode d'estimation des plus proches voisins, la version spatio-temporelle de PatchMatch proposée par Newson et al. [2] est adoptée avec deux modifications importantes pour améliorer son efficacité :

— La première modification accélére PatchMatch en parallélisant l'algorithme suivant la technique "jump flood" de Barnes et al.[3]. Une autre modification pour réduire le temps de calcul est d'utiliser une grille non-dense pour l'étape de recherche aléatoire. Il s'avère en effet qu'il n'est pas nécessaire de faire une étape de recherche aléatoire pour chaque pixel inconnu. La combinaison de ces deux facteurs offre à notre algorithme un gain de l'ordre de 5 à 7 pour la même précision.
— La deuxième modification concerne l'étape de propagation dans la direction temporelle, qui est guidée par le flot optique. L'étape de propagation de l'algorithme PatchMatch repose sur l'hypothèse que des patches voisins ont des plus proches voisins similaires. Pour la cohérence temporelle, cette hypothèse n'est valide que si le fond est statique ou en mouvement périodique. Pour conserver la cohérence temporelle, nous propageons l'information de plus proche voisin suivant la direction du vecteur de flot optique. Formellement, dans l'étape de propagation, le voisinage temporel pour le centre du patch au pixel $(x, y, t), P_{(x,y,t)}$ est $P_{(x+O_x, y+O_y, T+1)}$ au lieu de $P_{(x,y,t+1)}$, où O_x et O_y représentent le flot optique dans les directions x et y.

2.5 Reconstruction

Dans cette étape, tous les pixels dans l'occultation sont reconstruits en utilisant une moyenne pondérée des valeurs données par les plus proches voisins, c'est-à-dire :

$$u(p) = \frac{\sum_{q \in N_p} s_q^p u\left(p + \phi(q)\right)}{\sum_{q \in N_p} s_q^p},$$

où les poids s_p^q sont définis comme :

$$s_q^p = \exp\left(-\frac{d^2\left(W_q^u, W_{q+\phi(q)}^u\right)}{2\sigma_p^2}\right) \psi_q \varphi_q^p.$$

Dans cette expression, le premier terme est classique [1, 2] et dépend de la similarité entre patches. Il est modulé par deux (nouveaux) termes multiplicatifs ψ_q et φ_q^p. Le premier est un paramètre de confiance défini comme :

$$\psi_q = \begin{cases} (1 - C_0) \exp\left(-\frac{d(q, \mathcal{H}^c)}{\sigma^2}\right) + C_0 & \text{si } q \in \mathcal{H} \\ 1 & \text{sinon,} \end{cases}$$

où $d(q, \mathcal{H}^c)$ est la distance entre le pixel q et la frontière de l'occultation, et C_0, σ^2 sont deux paramètres. Ce terme ψ_q permet d'accorder plus de confiance aux patches situés près de la frontière de l'occultation de manière à guider l'information de la frontière vers le centre, ce qui permet d'éliminer certains artefacts à la frontière. Quant au terme φ_q^p, il s'agit d'un paramètre de séparation défini comme suit :

$$\varphi_p^q = \begin{cases} 1 & \text{si } p, q \text{ sont du même type "fond" ou "objet mobile"} \\ 0 & \text{si } p, q \text{ sont de types différents} \end{cases}$$

Pour classifier les pixels en "fond" ou "objet mobile", on utilise un seuillage du flot optique. Ce terme φ_q^p offre un moyen simple d'éviter les effets de mélange entre le fond et les objets en mouvement dans le résultat final.

3 Résultats

Nous avons évalué notre méthode pour différentes séquences vidéo qui présentent des objets recouverts par une occultation fixe ou mobile, une instabilité de camera, un fond dynamique, une grande région à reconstruire, etc. Nous comparons les reconstructions obtenues avec deux méthodes récentes, [2] et [5], en utilisant leurs bases de données. Les resultats sont disponibles à l'adresse `https://purl.org/vid_inp_motion`.

(a) (b)

FIGURE 1 – (a) Quelques images représentatives des résultats obtenus avec notre approche. De haut en bas : séquences `tennis`, `bxm-bump` et `elephant`. (b) Pour le profil x correspondant au segment rouge dans l'image du haut, le profil reconstruit $x-t$ est représenté dans les deux images du bas (avec, puis sans la frontière de la région à reconstruire).

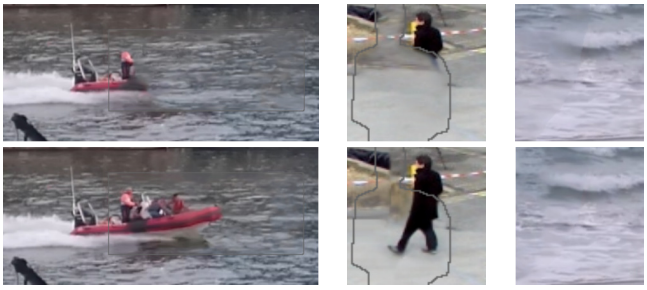
3.1 Suppression des objets indésirables

Le but de cette expérience est de supprimer les objets indésirables dans des vidéos enregistrées à l'aide d'une caméra mobile. La même base de données que celle retenue dans Huang et al. [5] est utilisée. Quelques images représentatives des résultats obtenus sont montrées sur la figure 1 (a), qui illustre la capacité de notre approche à bien préserver la structure spatiale. La figure 1 (b) détaille la reconstruction d'un profil $x-t$, où x décrit le segment rouge tracé sur une image de la séquence `elephant`. Cet exemple illustre la bonne préservation de la cohérence temporelle, qui résulte de la combinaison des patches spatio-temporels 3D et du flot optique. Pour cet exemple précis, la cohérence temporelle est également respectée en utilisant la méthode de Huang et al. [5]. Cependant, cette dernière méthode se limite uniquement à des patches 2D, et la qualité de la cohérence temporelle dépend donc de manière critique de la précision du flot optique, ce qui peut conduire à des erreurs grossières dans certaines séquences telles que `Mallard` ou `Breakdance`. De plus, la synthèse incorrecte du flot optique peut conduire à des artefacts désagréables. Ceci est par exemple visible dans l'article [5] sur la séquence `Loulous`, séquence pour laquelle notre méthode fournit un résultat plausible.

Un autre avantage de notre méthode est la réduction du temps de calcul. Alors que la méthode de Huang et al. [5], qui n'utilise pourtant que des patches 2D, requiert environ 3 heures pour compléter une vidéo (par exemple la séquence `bear`), notre méthode ne nécessite qu'environ 30 minutes.

3.2 Reconstruction d'objets en mouvement

Un autre cas de figure intéressant, et difficile, est celui des vidéos dans lesquelles un objet en mouvement traverse une occultation fixe ou mobile. La figure 2 montre quelques exemples de reconstruction, pour lesquels notre approche donne des résultats meilleurs que celle de Newson et al. [2]. Ainsi, les figures 2 (a) et (b) montrent que la méthode proposée dans [2] est incapable de reconstruire l'objet mobile (le bateau ou le piéton), alors que notre algorithme reconstruit à la fois le fond et



(a) (b) (c)

FIGURE 2 – Quelques trames (recadrées) de résultats pour les séquences suivantes : (a) `Bateau`, (b) `S2L1`, (c) `Loulous`. En haut : résultat de Newson et al. [2], en bas : notre résultat.

l'objet en mouvement. De plus, en intégrant la carte de confiance dans l'étape de reconstruction, notre résultat produit moins d'artefacts à la frontière de la zone à reconstruire par rapport à la méthode de Newson et al. [2], comme l'illustre la figure 2 (c).

4 Conclusion

Cet article présente une technique d'inpainting vidéo qui améliore l'état de l'art sur deux points importants : la capacité à reproduire les mouvements, potentiellement complexes et sur une longue durée, et une diminution du temps de calcul. Ces progrès sont obtenus en ayant fortement recours au flot optique et grâce à la parallélisation de l'algorithme.

Références

[1] Yonatan Wexler, Eli Shechtman, and Michal Irani, "Space-time completion of video," *IEEE Transactions on pattern analysis and machine intelligence*, 2007.

[2] Alasdair Newson, Andrés Almansa, Matthieu Fradet, Yann Gousseau, and Patrick Pérez, "Video inpainting of complex scenes," *SIAM Journal on Imaging Sciences*, vol. 7, no. 4, pp. 1993–2019, 2014.

[3] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan Goldman, "Patchmatch : a randomized correspondence algorithm for structural image editing," *ACM Transactions on Graphics-TOG*, vol. 28, no. 3, pp. 24, 2009.

[4] Miguel Granados, James Tompkin, K Kim, Oliver Grau, Jan Kautz, and Christian Theobalt, "How not to be seen—object removal from videos of crowded scenes," in *Computer Graphics Forum*. Wiley Online Library, 2012, vol. 31, pp. 219–228.

[5] Jia-Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf, "Temporally coherent completion of dynamic video," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, pp. 196, 2016.

[6] Nobuyuki Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.