

# Restauration de cartes de profondeur basée sur les superpixels pour la synthèse de vue RGB-D

Pierre BUYSSENS, Maxime DAISY, David TSCHUMPERLÉ, Olivier LÉZORAY

GREYC CNRS (UMR 6072), UNICAEN, ENSICAEN, Équipe Image  
6, Bd. Maréchal Juin, 14000 Caen, FRANCE  
{prénom.nom}@ensicaen.fr

**Résumé** – Nous proposons une méthode pour restaurer les trous d’une carte de profondeur qui apparaissent lors de la synthèse de vues virtuelles à partir de scènes RGB-D. Basée sur une sursegmentation en superpixels des vues originales et synthétisées, l’approche proposée gère efficacement de nombreuses occlusions où la plupart des approches existantes échouent. L’utilisation des superpixels rend l’algorithme plus robuste tout en permettant de modéliser efficacement l’image. De nombreuses comparaisons à l’état de l’art montrent l’efficacité de notre approche.

**Abstract** – In this paper we propose an approach to inpaint holes in depth maps that appear when synthesizing virtual views from a RGB-D scenes. Based on a superpixel oversegmentation of both the original and synthesized views, the proposed approach efficiently deals with many occlusion situations where most of previous approaches fail. The use of superpixels makes the algorithm more robust to inaccurate depth maps, while giving an efficient way to model the image. Extensive comparisons to relevant state-of-the-art methods show that our approach outperforms qualitatively and quantitatively these existing approaches.

## 1 Introduction

La télévision 3D et le cadre plus général du rendu de scène d’un point de vue quelconque sont devenus des technologies prometteuses dans le domaine de la 3D. Afin de synthétiser des vues virtuelles à partir de vues connues, la technique la plus utilisée consiste à utiliser une carte de profondeur qui a été générée en même temps que l’image d’intensité. À l’aide de ces deux images, une vue virtuelle d’une scène à partir d’un nouveau point de vue peut alors être synthétisée en déformant ces images (*warping*) [5].

Un problème important se pose alors avec l’apparition d’occlusions : les parties du fond qui sont cachées (et inconnues) par un objet au premier plan dans l’image originale peuvent devoir être rendues dans la vue synthétisée (Fig. 1). La tâche consistant à remplir ces trous est appelée *désocclusion* et fait partie du problème plus général de la *restauration* ou *inpainting*. Cependant et contrairement au problème général de la suppression d’un objet d’une scène, plusieurs remarques peuvent être faites ici : étant donné que les occlusions sont dues à une différence significative de profondeur entre deux objets, elle sont presque toujours entourées par du fond et des objets au premier plan. De plus, la carte de profondeur peut avoir un rôle crucial pour guider la restauration [4].

**Vue d’ensemble de l’état de l’art** : de nombreux travaux de l’état de l’art [4, 6, 1, 9] s’attaquent au problème de désocclusion en se basant sur des méthodes éprouvées de restauration. À ce titre, l’algorithme de Criminisi *et al.* [3] est largement utilisé pour son efficacité et sa capacité à reconstruire de grandes

portions d’image.

La plupart de ces travaux se concentrent sur l’*inpainting* de l’image d’intensité. Le sous-problème (pas si facile) de l’*inpainting* de la carte de profondeur est ainsi souvent traité à l’aide de méthodes triviales, bien souvent insuffisantes. De plus, seuls [6] et [9] traitent spécifiquement ce problème avant la restauration de l’image d’intensités.

**Problématique** : Un problème majeur arrive lorsqu’une occlusion qui doit être restaurée avec du fond n’est entourée que d’objets du premier plan. La figure 1 illustre ce cas : étant donnée la déformation pour la création de la vue synthétique, le trou pointé par la flèche rouge ne peut être restauré qu’avec les pixels voisins connus. Ne considérant que ces voisins les plus proches, de nombreuses approches de la littérature échouent à restaurer une telle occlusion. À notre connaissance, seule l’approche proposée dans [9] résout (partiellement) ce problème.

L’idée de base de [9] pour la restauration de la carte de profondeur est d’utiliser la vue originale (avant déformation) pour inférer correctement les valeurs de profondeur (fond ou premier plan). Plus spécifiquement, la méthode *scanline* de [9] analyse les distributions de profondeur des bords des trous des images originale et synthétisée. Les maxima locaux de ces distributions sont ensuite utilisés pour inférer la profondeur de la ligne servant à restaurer la carte de profondeur. Cette algorithme souffre cependant de plusieurs défaut :

- étant donné que la restauration se base sur des lignes horizontales (de profondeur constante), un plan appartenant au fond qui n’est pas parallèle au plan de la caméra ne peut être correctement restauré,



FIGURE 1 – Illustration d’un problème de restauration de fond : Le trou (pointé par la flèche rouge) n’est entouré que de pixels de premier plan. Haut : image d’intensité synthétisée, milieu : carte de profondeur synthétisée, bas : carte de profondeur restaurée avec notre méthode.

- l’utilisation de lignes horizontales peut éventuellement suffire lors d’une translation du point de vue, mais devient clairement insuffisante pour des déformations plus générales (rotation...).

**Contributions :** Cet article se concentre sur la restauration des occlusions qui apparaissent dans une carte de profondeur suite à une déformation. Basé sur une idée similaire à [9], l’algorithme proposé utilise l’image originale (avant déformation) pour restaurer correctement les valeurs des pixels masqués. Détaillé à la section 2, notre algorithme est comparé à la fois qualitativement et quantitativement à [9] dans la section 3.

## 2 Méthode de restauration de cartes de profondeur

La solution proposée repose sur la modélisation par des plans des superpixels de l’image originale et de leur combinaison pour restaurer les trous.

L’algorithme proposé est composé de 3 étapes :

- 1) calcul des sursegmentations denses en superpixels des images originale et synthétisée, et des correspondances entre superpixels des deux images,
- 2) pour chaque pixel à restaurer, trouver un ensemble de superpixels candidats dans l’image originale,
- 3) modéliser ces superpixels par des plans et restaurer la profondeur des pixels via des combinaisons linéaires de ces plans.

**Notations :** une carte de profondeur source  $S$  est déformée pour synthétiser une nouvelle vue de profondeur  $D$ . Les pixels  $p \in D$  à restaurer forment un ensemble de trous  $\Omega = \{\Omega_1 \dots \Omega_n\}$ .

Les superpixels  $R_i^S$  et  $R_i^D$  désignent les superpixels partageant le même label  $i$  et appartenant à  $S$  et à  $D$ . Dans la suite, nous adoptons la convention qu’un pixel qui est loin de la caméra a une profondeur faible, tandis qu’un pixel proche de la caméra a une profondeur élevée (disparité).

### 2.1 Calcul des superpixels

Notre méthode commence par calculer une sur-segmentation de l’image source  $S$  en superpixels. Pour cela, nous utilisons l’algorithme *Eikonal-based Region Growing Clustering* (ERGC) [2] qui formule le regroupement des pixels comme la solution d’une équation Eikonale. La carte de labels produite est ensuite déformée vers la nouvelle vue [7], et érodée par un élément structurant circulaire de rayon 1. Cette érosion corrige les mauvais labels qui peuvent être donnés à certains pixels appartenant aux bords des objets. ERGC est ensuite appliquée sur  $D$  en prenant les superpixels érodés comme germes, sans diffuser dans les occlusions. La figure 2 montre quelques correspondances entre superpixels de  $S$  et  $D$ .

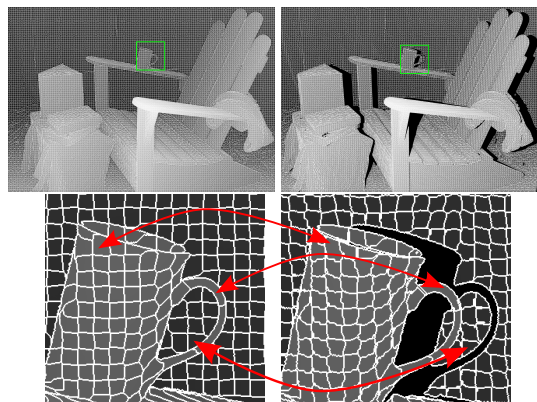


FIGURE 2 – Illustration de la correspondance entre superpixels de  $S$  (gauche) et  $D$  (droite).

### 2.2 Recherche des plans sources

Soit un pixel  $p$  à restaurer, la recherche des plans sources consiste à trouver un ensemble de superpixels qui vont être utilisés pour restaurer la profondeur de  $p$ , et est décomposée en 3 étapes :

- 1) Le pixel  $q_1 \in D \setminus \Omega$  le plus proche de  $p$  est trouvé via une descente de gradient sur la fonction de distances des pixels  $p_i \in \Omega$  aux pixels  $p_j \in D \setminus \Omega$ . Le superpixel correspondant est noté  $R_1^D$ . Un deuxième pixel  $q_2 \in D \setminus \Omega$  est également considéré. Ce pixel est celui se trouvant sur le bord opposé du trou est et trouvé en suivant la ligne définie par  $(q_1, p)$ . Le superpixel correspondant est noté  $R_2^D$ . Sont ainsi définis les offsets  $\Delta_1^p = \|p - q_1\|$  et  $\Delta_2^p = \|p - q_2\|$ .
- 2)  $q_1$  et  $q_2$  sont ensuite reportés sur  $S$  à l’aide de la déformation inverse, avec la contrainte qu’ils appartiennent respectivement à  $R_1^S$  et  $R_2^S$ . À partir de ces deux nouveaux pixels  $q_3 \in S$  et  $q_4 \in S$ , les offsets  $-\Delta_1^p$  et  $-\Delta_2^p$  pointent vers deux nouveaux superpixels  $R_3^S$  et  $R_4^S$ . À noter que  $R_3^S$  et  $R_4^S$  peuvent être un seul et même superpixel.

3) De l'ensemble des superpixels  $\{R_1^D, R_2^D, R_3^S, R_4^S\}$ , seuls ceux ayant la profondeur moyenne la plus faible (à un epsilon près) sont gardés et forment l'ensemble des superpixels utilisés ensuite pour la restauration de la profondeur de  $p$ .

La figure 3 illustre le processus :  $R_1^D$  et  $R_2^D$  sont les superpixels colorés en bleu et rouge (deuxième et troisième lignes).  $R_3^S$  et  $R_4^S$  peuvent alors être deux superpixels différents (bleu sombre et rouge sombre, troisième ligne) ou peuvent être identiques (superpixel vert de la deuxième ligne).

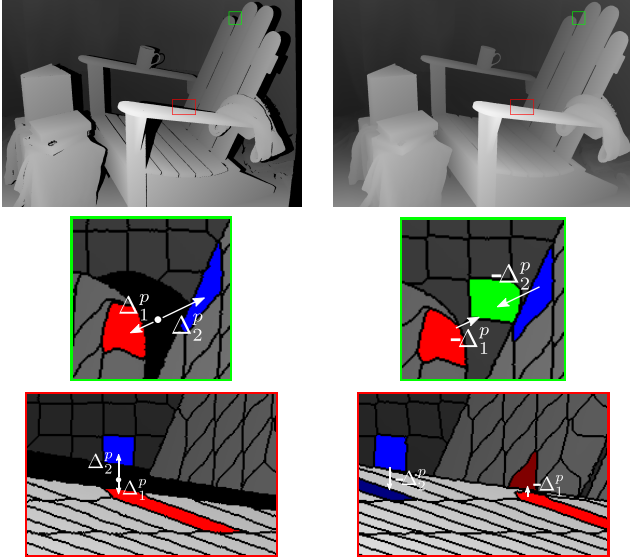


FIGURE 3 – Recherche des superpixels candidats pour la restauration de la profondeur du pixel  $p$  (point blanc). Les superpixels rouge et bleu sont trouvés dans (voir la section 2.2) dans  $S$  (colonne de gauche). Les offsets  $-\Delta_1^p$  et  $-\Delta_2^p$  sont reportés sur l'image originale (colonne de droite) et peuvent pointer vers le même superpixel (coloré en vert, deuxième ligne) ou vers des superpixels différents (colorés en bleu et rouge sombres, troisième ligne).

### 2.3 Restauration de la profondeur

Chaque superpixel candidat à la restauration est modélisé par un plan par la méthode des moindres carrés. La profondeur de  $p$  est alors calculée comme étant la combinaison linéaire de ces plans. Le poids associé à chaque plan est inversement proportionnel à la distance de  $p$  à chacun de ces superpixels.

### 2.4 Discussion

L'utilisation des superpixels présente deux avantages :

- Étant donné que la déformation peut ne pas être précise au pixel près, la déformation d'un seul pixel peut conduire à des erreurs. L'utilisation de superpixels rend l'algorithme plus robuste, particulièrement lors de la recherche des pixels  $q_3$  et  $q_4$ .
- Les superpixels issus de ERGC offrent une bonne adhérence aux contours des objets et définissent une sursegmentation de la carte de profondeur en régions homogènes. Les plans modélisant ces superpixels sont ainsi suffisants pour la restauration des profondeurs manquantes.

**Complexité :** La complexité de l'approche proposée repose essentiellement sur le calcul des superpixels de  $S$ . ERGC utilise l'algorithme du Fast-Marching qui requiert le tri des pixels selon leur distance géodésique au germe le plus proche, d'où une complexité globale en  $\mathcal{O}(n \log n)$ . Malgré cette complexité théorique, ERGC est quasi-linéaire en pratique. Le reste de l'algorithme est linéaire en temps et peut de plus être facilement parallélisé étant donné que les pixels sont traités indépendamment. Pour information, sans parallélisation, la restauration des 526389 pixels de la figure 3 est effectuée en environ 15s.

## 3 Résultats

**Comparaisons sur une image synthétique :** Nous montrons ici des résultats de restauration sur une image synthétique composée de deux objets : un disque perforé de deux trous circulaires devant un plan non parallèle au plan de la caméra (voir la figure 4a). En changeant le point de vue de la scène, de nombreuses occlusions apparaissent, et en particulier un trou circulaire qui n'est entouré que par des pixels appartenant au disque au premier plan (Figure 4b et 4c). Tandis que la méthode [6] échoue à replir ce trou circulaire avec des valeurs de profondeur de fond (Fig. 4d), [9] échoue à restaurer correctement les profondeurs du plan de fond (Fig. 4e). Notre approche restaure correctement cette carte de profondeur.

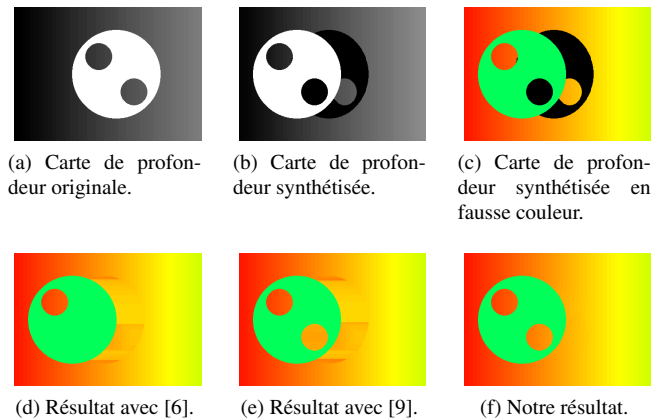


FIGURE 4 – Restauration d'une carte de profondeur synthétique (en fausses couleurs pour la visualisation).

**Comparaisons sur des données réelles :** Nous comparons ici notre méthode avec [9] sur la base de données d'images stéréo Middlebury-2014 [8], composée de 23 paires d'images ( $\sim 5.5$ Mpixels) pour lesquelles la carte de profondeur est connue (vérité terrain). La synthèse s'effectue de la vue 0 à la vue 1.

La figure 5 compare la restauration d'une carte de profondeur obtenue à l'aide de notre méthode à celle proposée dans [9]. Cette comparaison met particulièrement en lumière les limitations des méthodes scanline, ainsi que les difficultés restantes pour la reconstruction des structures des objets.

La figure 6 compare pour chaque image de la base les erreurs

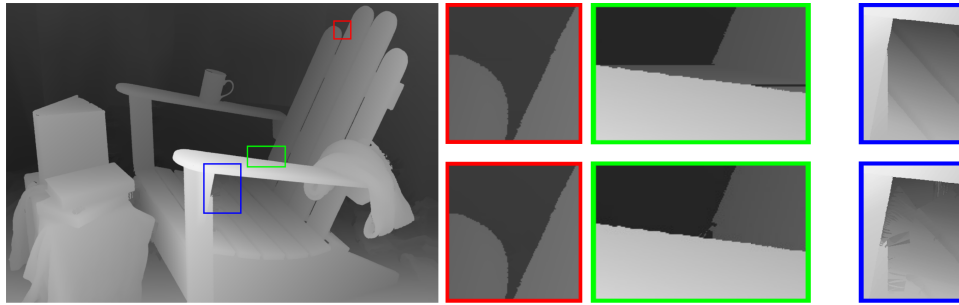


FIGURE 5 – Première colonne : carte de profondeur restaurée avec notre méthode. Colonnes 2 et 3 : comparaisons de la restauration avec notre méthode (bas) et celle proposée dans [9] (haut). Dernière colonne, comparaison de la restauration avec notre méthode (bas) et la vérité terrain (haut). Notre méthode présente des défauts lors de la reconstruction des structures.

moyennes et médianes de la restauration par la méthode [9] et la nôtre. Ces résultats expérimentaux montrent la nette amélioration de notre méthode sur [9], et soulignent l’efficacité de la méthode que nous proposons.

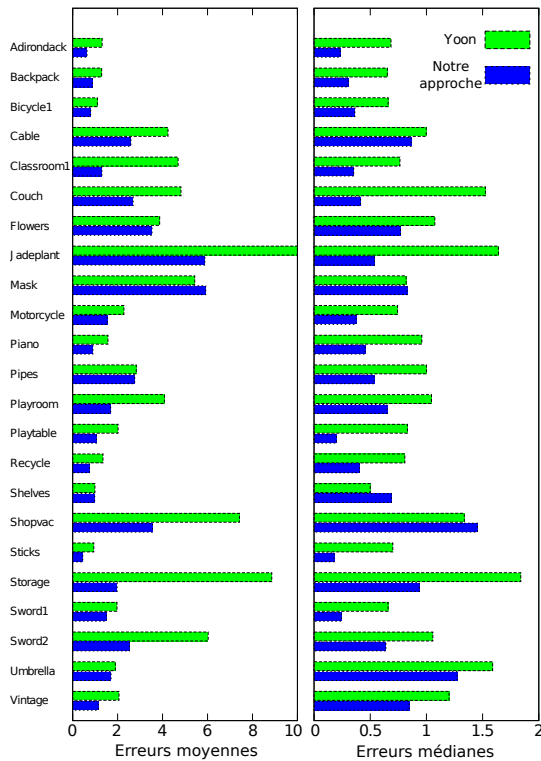


FIGURE 6 – Erreurs moyennes et médianes pour chaque image de la base Middlebury-2014.

## 4 Conclusion

Dans cet article nous avons proposé une nouvelle méthode pour traiter spécifiquement la restauration des occlusions de cartes de profondeurs synthétisées. Basée sur des superpixels, l’approche proposée restaure nettement mieux les cartes de profondeur que les approches existantes de la littérature. Fort de ces cartes de profondeurs restaurées, nos futurs travaux porteront sur leur utilisation pour la restauration des images d’inten-

sité synthétisées.

## Références

- [1] I. Ahn and C. Kim. Depth-based disocclusion filling for virtual view synthesis. In *IEEE International Conference on Multimedia and Expo*, pages 109–114, 2012.
- [2] P. Buysse, I. Gardin, S. Ruan, and A. Elmoataz. Eikonal-Based region growing for efficient clustering. *Image and Vision Computing*, 32(12) :1045–1054, 2014.
- [3] A. Criminisi, P. Pérez, and K. Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing*, 13(9) :1200–1212, 2004.
- [4] I. Daribo and B. Pesquet-Popescu. Depth-aided image inpainting for novel view synthesis. In *IEEE International Workshop on Multimedia Signal Processing*, pages 167–170. IEEE, 2010.
- [5] C. Fehn. Depth-image-based rendering (dibr), compression, and transmission for a new approach on 3d-tv. In *Electronic Imaging*, pages 93–104. International Society for Optics and Photonics, 2004.
- [6] P. Ndjiki-Nya, M. Koppel, D. Doshkov, H. Lakshman, P. Merkle, K. Muller, and T. Wiegand. Depth image-based rendering with advanced texture synthesis for 3-d video. *Multimedia, IEEE Transactions on*, 13(3) :453–465, 2011.
- [7] K.-J. Oh, S. Yea, and Y.-S. Ho. Hole filling method using depth based in-painting for view synthesis in free viewpoint television and 3-d video. In *Picture Coding Symposium, 2009. PCS 2009*, pages 1–4. IEEE, 2009.
- [8] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nešić, X. Wang, and P. Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *Pattern Recognition*, pages 31–42. Springer, 2014.
- [9] S. S. Yoon, H. Sohn, and Y. M. Ro. Inter-view consistent hole filling in view extrapolation for multi-view image generation. In *ICIP*. 2014.