

# Recalage de masses de données 3D/couleur non structurées

Sébastien DRUON, André CROSNIER et Marie-José ALDON

<sup>1</sup> LIRMM, UMR 5506 Université Montpellier II - CNRS

<sup>2</sup> 161 rue Ada, 34000 Montpellier - France

<sup>3</sup> {druon, crosnier, aldon}@lirmm.fr

## Résumé

*Cette étude concerne le recalage de nuages de points denses et non structurés pour la construction de modèles 3D/couleur d'objets complexes. Notre objectif est d'accroître les performances de l'algorithme ICP classique en utilisant l'information couleur attachée aux points, de manière à pouvoir traiter de grandes masses de données, mais aussi recalculer précisément des données issues d'objets pour lesquels l'information géométrique n'est pas discriminante. Après avoir rappelé brièvement le principe du recalage itératif basé sur la recherche du point le plus proche dans l'espace 3D (Iterative Closest Point) et des travaux de recherche réalisés dans ce domaine, nous proposons une nouvelle variante de cet algorithme qui permet d'améliorer la sélection des points. L'information couleur y est utilisée comme une contrainte pour réduire la taille de l'espace de recherche durant la phase de mise en correspondance. La validation expérimentale est réalisée avec les images 3D/couleur résultant de la numérisation à haute définition et sous divers points de vue de trois types d'objets.*

## Mots clefs

Images 3D/couleur, recalage, mise en correspondance.

## 1 Introduction

La construction automatique de modèles d'objets ou de scènes 3D ou 3D texturés est un problème récurrent pour des applications nombreuses et diversifiées. En effet, ces modèles peuvent être utilisés pour représenter l'environnement dans lequel navigue un robot, pour construire des scènes de réalité virtuelle à partir de l'observation d'objets réels, pour représenter des monuments historiques qui ont été numérisés en vue de leur restauration, ou encore pour alimenter une base de données d'objets de musée consultable par un large public. La construction d'un modèle 3D comporte trois étapes : l'acquisition d'un ensemble d'images 3D sous différents points de vue, leur recalage dans un repère unique, la simplification des données obtenues pour éliminer les redondances résultant du recouvrement des

images et le calcul d'une représentation surfacique de l'objet.

Deux grandes catégories de méthodes sont utilisées pour acquérir les données. La première qui est basée sur la vision active en 3D (mesure du temps de vol, vision en lumière structurée, triangulation laser) fournit des images denses de la scène observée (images de distance et parfois images vidéo). La deuxième catégorie qui fait appel à une ou plusieurs caméras vidéo (stéréovision, vision en mouvement) permet d'acquérir la texture de la scène ainsi qu'une structure 3D minimale (par exemple, un ensemble de points d'intérêt). Mais en règle générale, on ne dispose pas d'une image 3D dense décrivant précisément la géométrie de la scène. Pour numériser complètement un objet ou une scène 3D il faut acquérir, sous des points de vue différents, un ensemble d'images qui se recouvrent partiellement. Le nombre de vues nécessaires dépend de la complexité et de la taille de l'objet, mais également de la résolution du capteur. On obtient donc un ensemble d'images qu'il faut recalculer. Ainsi, un scanner 3D fournit des nuages de points non structurés qu'il faut ramener dans un repère unique. Lorsque ces images sont recalées deux par deux on parle de recalage simple, lorsqu'elles sont recalées toutes ensemble, il s'agit d'un recalage global [1].

Ce travail de recherche est réalisé dans le cadre de la création de modèles 3D/couleur à haute résolution d'objets de musée (peintures à l'huile, sculptures, figurines archéologiques, ...). Dans ce but, nous développons une approche intégrée pour la construction de modèles 3D texturés qui tire parti du caractère complémentaire des données vidéo et 3D fournies par les scanners à lumière structurée. Ce type de capteur présente l'avantage de fournir simultanément ces deux types de données dans le même repère (repère attaché à la caméra). Le sujet de cette publication est le recalage de deux nuages de points 3D/couleur denses et non structurés. Notre objectif est d'améliorer les performances de l'algorithme ICP (Iterative Closest Point) classique [2,3] en exploitant les données couleur attachées aux points 3D fournis par le capteur, afin d'être en mesure de traiter :

- de grands ensembles de données (plusieurs millions de points par nuage),

- des objets pour lesquels l'information géométrique n'est pas suffisante (relief peu marqué, symétries conduisant à des solutions multiples, ...).

Le papier est organisé de la manière suivante. Dans la section suivante, nous présentons l'algorithme ICP utilisé pour recalcr des nuages de points 3D et les travaux de recherche développés pour améliorer ses performances. En section 3, nous proposons une nouvelle méthode qui utilise l'information couleur comme une contrainte dans la phase d'appariement des points 3D. La dernière partie décrit les résultats d'une validation expérimentale réalisée avec des images 3D/couleur issues de la numérisation à haute définition et sous divers points de vue de 3 types d'objets : une statue en bois peint, une boîte cylindrique en ivoire avec des motifs colorés, et une peinture à l'huile.

## 2 Etat de l'art

Recaler deux ensembles de données consiste à estimer la transformation rigide qui permet de les représenter dans un repère unique [4]. Les données à recalcr peuvent être de nature diverse : images d'intensité, images surfaciques, images volumiques ou images multimodales (par exemple : images de distance texturées). Nous présentons ci-après un aperçu des techniques de recalcr basées sur l'algorithme ICP qui sont applicables aux images de distance prises de différents points de vue pour numériser un objet ou une scène. Le recalcr itératif basé sur la recherche du point le plus proche (Iterative Closest Point) [2,3] est une des techniques les plus couramment utilisées pour aligner deux nuages de points dans l'espace 3D.

### 2.1 L'algorithme ICP

Cet algorithme estime la transformation de mouvement rigide entre les deux formes 3D en considérant que :

- elles se recouvrent partiellement, c'est à dire qu'un sous-ensemble de points est commun aux deux nuages,
- elles sont a priori partiellement alignées, autrement dit que l'on dispose d'une estimation de la transformation initiale.

L'algorithme comporte deux étapes : la première génère un ensemble de correspondances temporaires entre points des deux nuages, la deuxième estime la transformation rigide qui relie ces points appariés.

Considérons deux nuages de points  $\{p_i\}$  et  $\{p'_i\}$  qui représentent les images d'un même objet acquises par un capteur de distance sous deux points de vue différents. D'un point de vue géométrique, on peut considérer que le point  $p_i$  correspondant au point  $p'_i$ , son plus proche voisin dans  $\{p_i\}$ , peut être défini par :

$$p_i = \arg \left[ \min_{p_j \in \{p_i\}} \|p_j - p'_i\| \right] \quad (1)$$

Pour chaque paire de points mis en correspondance  $p_i$  et  $p'_i$ , nous devons estimer une matrice de rotation  $R$  (3x3) et un vecteur de translation  $t$  (3x1) tels que :

$$p_i = R p'_i + t \quad (2)$$

Pour simplifier l'écriture, nous utiliserons la notation en coordonnées homogènes qui fait appel à la matrice  $T$  fonction des 6 paramètres de mouvement. Théoriquement, 3 paires de points sont suffisantes pour identifier la transformation  $T$ . En pratique, compte tenu du bruit qui entache les données et de la présence de zones d'occultation dans les images, des erreurs d'appariement subsistent. Pour estimer les paramètres du mouvement, on fait appel à une technique de minimisation d'erreur. Généralement [5], on recherche la solution au sens des moindres carrés, d'un système surdéterminé de  $N$  équations ( $N > 3$ ).

Le critère à minimiser s'écrit :

$$\varepsilon = \frac{1}{N} \sum_{i=1}^N \|p_i - T p'_i\|^2 \quad (3)$$

Cette procédure en deux étapes est itérée jusqu'à la convergence, c'est à dire jusqu'à ce que la variation du critère (3) reste inférieure à un seuil fixé a priori en fonction de la précision souhaitée pour le recalcr. L'algorithme ICP peut être résumé de la façon suivante :

Soit $T_0$ une estimation de la transformation initiale
Répéter, pour $k = 1 \dots k_{\max}$ , ou jusqu'à ce que le critère soit atteint
Trouver un ensemble de $N_k$ paires de points les plus proches dans $\{p_i\}$ et $\{p'_i\}$ , selon (1);
Estimer la transformation $T_k$ qui minimise le critère de distance (3)
Appliquer la transformation $T_k$ à tous les points de $\{p'_i\}$ .

Besl et Mc Kay [2] montrent que l'algorithme ICP converge de manière monotone vers le minimum local le plus proche. Lorsque l'initialisation n'est pas assez précise, ce type de faux appariement est fréquent durant les premières itérations de l'algorithme et conduit généralement à une estimation biaisée.

### 2.2 Variantes de l'algorithme ICP

De nombreuses variantes ont été proposées pour améliorer l'algorithme ICP introduit par Besl et Mc Kay. Le lecteur trouvera dans [5] une classification et une comparaison expérimentale de ces différentes solutions. Nous donnons ici un aperçu des principales approches qui ont pour objectif de réduire le volume des calculs nécessaires pour la mise en correspondance et/ou de rendre les appariements plus robustes vis à vis des différentes sources de bruit et de l'imprécision de l'initialisation. Alors que dans [2] tous les points disponibles sont utilisés dans la phase d'appariement, beaucoup de méthodes

n'utilisent qu'une partie des points de l'une ou des deux images. Ainsi, dans [7] un tirage aléatoire permet d'utiliser des échantillons de points différents à chaque itération. Un k-D tree est parfois mis en œuvre pour accélérer la recherche des points les plus proches [9].

Des attributs invariants par rapport au point de vue peuvent être utilisés pour caractériser les points durant la phase d'appariement. Ils peuvent être calculés soit à partir de l'image de distance, soit à partir d'informations complémentaires fournies par le capteur de numérisation (intensité, couleur, ...). Dans [8], un vecteur d'attributs géométriques et photométriques est attaché à chaque point de l'image. Il inclut les amplitudes des courbures principales calculées à partir de l'image de distance, ainsi que des données photométriques extraites de l'image couleur (coefficients de réflexion diffuse). Des données photométriques sont également exploitées dans [10] et [11]. Parmi les autres types de descripteurs on trouve : la courbure locale, la texture [9], la forme (attributs différentiels 3D) [7] et l'angle entre les normales [11]. Différentes solutions ont été explorées pour éliminer les faux appariements qui empêchent l'estimateur de converger vers la bonne solution. Dans [9], Zhang élimine les paires de points dont la distance est supérieure à un seuil qui est remis à jour de manière adaptative en fonction des paramètres statistiques de la distribution des distances obtenue au pas précédent. Dans [7], un estimateur LMS (Least Mean Square) minimise la médiane des carrés des résidus et permet de classer les points 3D de chaque image en fonction de la qualité de l'appariement qu'ils génèrent. Les points occultés, mal appariés ou qui n'ont pas de correspondant sont ainsi éliminés de la liste qui sera utilisée pour l'estimation de la transformation. Une autre solution pour accroître la robustesse du processus d'appariement consiste à rajouter des termes supplémentaires qui mesurent la similarité des points dans la fonction de distance (5). Ainsi, dans [12] une distance dans l'espace couleur est rajoutée à la distance dans l'espace 3D. La difficulté dans cette approche est le choix des coefficients de pondération affectés à chacun des termes.

Nous proposons dans cette publication une approche nouvelle pour exploiter la couleur dans l'appariement d'images 3D/couleur. Elle est basée sur une classification préliminaire des données en fonction de leur teinte, afin de sélectionner un nombre réduit de points pour réaliser la mise en correspondance.

### 3 Appariement 3D contraint

La contribution de cette recherche porte donc essentiellement sur l'utilisation de la couleur pour sélectionner les points à apparier. L'approche utilisée consiste à classer les points en fonction de leur couleur, en ayant pour objectifs :

- d'améliorer la robustesse de la mise en correspondance en autorisant uniquement l'appariement de points de couleur compatible,
- de réduire le nombre de données à traiter,
- de traiter le cas d'objets pour lequel le recalage ne peut pas être résolu en utilisant uniquement l'information 3D (formes symétriques, relief assez plat, ...)

#### 3.1 Classification basée sur la couleur

La segmentation de chaque nuage est effectuée à partir de la teinte  $H$  des points calculée selon la définition de l'espace HSV proposée dans [13]. La saturation  $S$  et l'intensité  $V$  ne sont pas considérées car elles varient avec les conditions d'acquisition. En effet, lorsque la numérisation est réalisée en éclairage ambiant, la variation de couleur d'un objet 3D provient essentiellement des variations de zones d'ombre entre les différents points de vue. Ces variations affectent seulement l'intensité des points, mais pas leur couleur intrinsèque. De plus, l'utilisation de la teinte comme critère de sélection des points ne nécessite pas de modèle d'illumination comme dans [8] où les auteurs calculent les paramètres de réflexion de la surface.

Nous considérons un espace couleur où la teinte  $H$  varie de 0 à 6, chaque valeur entière étant associée à une couleur pure. Une classe couleur est caractérisée par un petit intervalle de teinte centré sur une de ces valeurs. Pour chaque nuage de points, nous générons les six sous-nuages correspondant à chaque classe, les points restants n'étant pas pris en considération. Lorsque la saturation  $S$  est inférieure à un seuil, la composante teinte ne semble pas significative, c'est pourquoi nous filtrons la composante  $S$ .

#### 3.2 Critère pour la sélection des classes

Actuellement l'algorithme de recalage utilise uniquement les points appartenant à une seule classe  $C$ , choisie en fonction de deux critères :

- Le ratio  $R_c$  de points contenus dans la classe. Une bonne classe doit avoir un nombre raisonnable de points. Par exemple, dans les expérimentations qui suivent, nous avons :

$$5\% < R_c < 15\% \quad (4)$$

Si une classe contient moins de 5% des données initiales, nous considérons que c'est un mauvais représentant de l'ensemble des données. La limite supérieure permet de réduire le temps de calcul en favorisant la sélection de régions d'intérêt de faible taille.

- Le ratio  $N_c$  de points de la classe  $C$  initialement inclus dans la boîte englobante du nuage 2. Ce critère est destiné à favoriser les classes incluses dans la zone de recouvrement des deux nuages. Etant donné que nous n'avons aucune information sur cette zone, nous

utilisons la transformation initiale fournie par l'utilisateur, et considérons que l'intersection entre les deux boîtes englobantes doit être représentative de cette région. Une classe possédant un grand nombre de points dans cette intersection est considérée comme plus favorable à un bon recalage.

Un score  $Sc$  est calculé pour chaque classe selon la relation :

$$Sc = Nc + f(Rc) \quad (5)$$

où  $f(Rc) = 100$  si (4) est respectée, 0 sinon.

### 3.3 Recalage itératif

Notre algorithme ICP utilise un seuil adaptatif  $Dmax$  pour éliminer les mauvais appariements, comme dans [9].

- Soit  $T_0$  l'estimation initiale de la transformation
- Initialiser  $Dmax$  à 20 fois la résolution des images
- Répéter pour  $k = 1 \dots k_{max}$  ou jusqu'à ce que le critère soit atteint
  - Initialiser à vide la liste  $L_k$  des paires retenues
  - Pour chaque point  $\{\bar{p}_i\}$  de la classe sélectionnée
    - Rechercher son plus proche voisin dans  $\{\bar{p}'_i\}$ .
    - Si  $distance(\bar{p}_i, \bar{p}'_i) < Dmax$ ,  
ajouter la paire à la liste  $L_k$  des paires enregistrées
  - Calculer la moyenne  $\mu$  et l'écart type  $\sigma$  des distances dans  $L_k$
  - Faire  $Dmax = \mu + \sigma$ 
    - Estimer la transformation  $T_k$  à partir des paires de  $L_k$
    - Appliquer la transformation  $T_k$  à tous les points de  $\{\bar{p}'_i\}$
- Répéter la boucle jusqu'à ce que la moyenne  $\mu$  et l'écart type  $\sigma$  soient stabilisés.

Pour accélérer le processus de recherche du plus proche voisin, on utilise un k-D tree, selon la méthode décrite dans [9]. La complexité de l'algorithme de recalage est :

$$n_1 \cdot n_2 \cdot \log(n_2)$$

où  $n_1$  et  $n_2$  sont les nombres de points de chaque nuage. Si  $k_{max}$  itérations sont exécutées sans atteindre la convergence, un nouveau recalage est exécuté avec la classe qui a le second meilleur score, et ainsi de suite.

## 4 Validation sur données réelles

Nous décrivons ici une évaluation de l'algorithme avec des données réelles provenant de la numérisation de trois types d'objets avec le système TRITOS développé par la société Breuckmann [14]. Ce capteur utilise une caméra couleur de 1280x1024 pixels, et un projecteur de franges en lumière blanche. Pour chaque point de mesure, il délivre les coordonnées 3D et les composantes RGB.

### 4.1 Description des données

Les jeux de données utilisées dans cette évaluation sont représentatifs de trois catégories d'objets de musée. Dans ces données, il existe un bruit sur la couleur du essentiellement à la numérisation (chaque composante couleur est codée seulement sur 5 bits), et aux variations d'éclairage pendant l'acquisition.

En ce qui concerne la qualité des données 3D, on peut remarquer que la statuette Bali (fig.4) a une forme suffisamment complexe pour générer un certain nombre de zones d'ombre pour le capteur. Par contre Ivoire (fig. 5) et Wallis (fig. 6) sont des exemples difficiles à traiter du fait de leur symétrie (cylindre et plan) et de l'absence de point d'intérêt 3D.

Nom de l'objet	Bali (fig 4)	Ivoire (fig 5)	Wallis (fig 6)
Nature	Statue en bois peint	Boite en ivoire colorée	Peinture à l'huile
Taille du nuage 1 (Nbre de points)	209628	312950	1030658
Taille du nuage 2	247461	306978	1036735
Résolution du capteur ( $\mu m$ ) X Y Z	180x180x6	60x60x3	60x60x3
Information 3D	Riche	Pauvre (cylindre)	Pauvre (plan)
Information couleur	Bruitée (reflets)	Riche	Riche

Table1 : Description des données

### 4.2 Résultats

Les résultats obtenus avec notre algorithme de recalage sont résumés dans la table 2. Pour chaque expérience, nous donnons, le nombre de points dans la classe sélectionnée pour chaque nuage, l'erreur moyenne finale en mm, le nombre d'itérations nécessaires pour atteindre la convergence, et le nombre final de paires de points dans  $L_k$ . Nous considérons que la convergence est atteinte quand les deux conditions suivantes sont vérifiées pour deux itérations successives de l'algorithme :

- le nombre de paires dans  $L_k$  reste constant,
- les variations de l'erreur moyenne et de l'écart type sont inférieures à 1e-6 mm.

On constate que, dans chaque expérience, l'erreur finale moyenne est proche de la résolution du capteur. Ceci confirme la qualité du recalage final.

Exécuté sur un Pentium Dell 4- 2,9 GHz – 1Go Ram, sans optimisation du code source, le recalage de deux nuages de points a demandé quelques minutes pour Ivoire et Bali. Par contre, le temps de calcul reste important pour Wallis. En effet, l'absence de relief significatif dans la direction Z se traduit par une structure de kD-tree dégénérée et donc un temps de mise en correspondance beaucoup plus long. Les courbes présentées dans la figure 1 montrent

l'évolution de la moyenne et de l'écart type des distances entre les nuages de points recalés, au cours des itérations successives lors du recalage de Bali. La figure 2 montre l'évolution du nombre de paires dans  $L_k$ .

Ces tests montrent qu'un recalage approximatif est obtenu au bout de 25 à 30 itérations. Les figures 3 à 6 visualisent les différents nuages de points 3D avant et après recalage. Il faut souligner que dans le cas d'Ivoire ou de Wallis, il serait impossible d'obtenir une convergence correcte avec un algorithme de recalage purement géométrique.

Nom de l'objet	Bali	Ivoire	Wallis
Classe sélectionnée	Rouge	Vert	Cyan
Pourcentage de points sélectionnés	5.95 %	7.06 %	9.10 %
Erreur moyenne (mm)	0.289311	0.06311	0.140817
Itérations	58	102	201
Nbre final de paires dans $L_k$	5677	11419	40473

Table 2 : Bilan des résultats

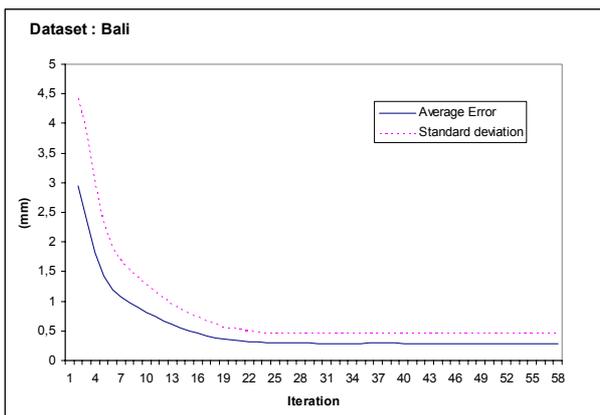


Figure 1 : Evolution de l'erreur de recalage avec Bali

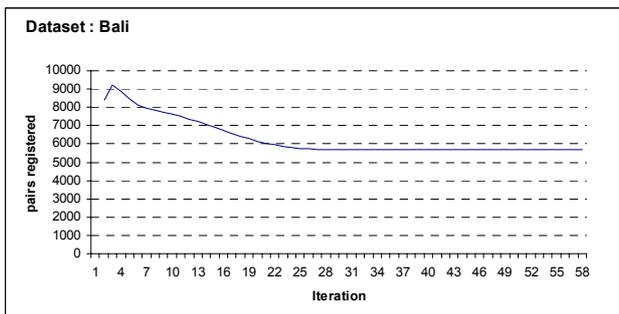


Figure 2 : Evolution du nombre de paires avec Bali

## 5 Conclusion

Nous avons proposé une nouvelle variante de l'algorithme ICP qui exploite l'information couleur pour permettre le recalage de grands ensembles de points 3D/couleur. Une segmentation préliminaire des données permet de

sélectionner des points appartenant à une ou plusieurs classes couleur intéressantes, de façon à accélérer le processus d'appariement 3D, et à augmenter la vitesse de convergence de l'algorithme de recalage itératif.

La validation expérimentale effectuée avec des données réelles montre que :

- le temps de calcul est réduit,
- la méthode fournit une estimation précise de la transformation rigide, y compris dans le cas où le recalage géométrique est difficile à résoudre avec un algorithme ICP classique.

Les améliorations en cours portent sur le processus de segmentation des données couleur. Un algorithme de clustering automatique est développé pour optimiser la sélection des classes. Par ailleurs, il est possible de réduire les temps de calcul en parallélisant un certain nombre de tâches. Nos travaux futurs porteront également sur une validation expérimentale plus complète, et en particulier sur une comparaison avec d'autres méthodes [7, 8, 9].

## 6 Références

- [1] R. Ben-Jemaa, F. Schmitt, "Recalage 3D", *Images de profondeur*, éditions Hermès, 2002.
- [2] P. Besl and N.D McKay, "A Method for Registration of 3-D Shapes", *Proc. of IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14 (2): 239-256, 1992
- [3] Y. Chen and G. Medioni, "Object Modelling by Registration of Multiple Range Images", *Image and Vision Computing*, 10 (3): 145-155, 1992.
- [4] S. Seeger, X. Laboureux, " Feature Extraction and Registration", *Principles of 3D Image Analysis and Synthesis*, Kluwer Academic Pub, 2000.
- [5] S. Rusinkiewicz and M. Levoy, "Efficient Variant of the ICP Algorithm". *Actes 3DIM*, pages 145-152, Canada, Québec, juin 2001.
- [6] G. Blais and M. D. Levine, "Registrating Multi-View Range Data to Create 3-D Computer Objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8) : 820-824, 1995.
- [7] T. Masuda and N. Yokoya, "A Robust Method for Registration and Segmentation of Multiple Range Images", *Computer Vision and Image Understanding*, 61(3): 295-307, mai 1995.
- [8] G. Godin, D. Laurendeau, R. Bergevin, " A method for the registration of Attributed Range Images", *Actes 3DIM*, pages 179-186, Québec, Canada, juin 2001,.
- [9] Z. Zhang, "Iterative Point Matching for Registration of Free Form Curves", *International Journal of Computer Vision*, 13 (2): 119-152, 1994.
- [10] S. Weik, "Registration of 3D partial surface models using luminance and depth information", *Actes 3DIM*, pages 93-100, Ottawa, Canada, mai 1997.
- [11] K. Pulli, "Multiview Registration for Large Data Sets", *Actes 3DIM*, pages 160-168, Ottawa, Canada, octobre 1999.
- [12] A. Johnson and S. Bing Kang, "Registration and Integration of Textured 3-D Data", *Actes 3DIM*, pages 234-241, Ottawa, Canada, mai 1997.
- [13] J.D. Foley, A. van Dam, S.K. Feiner, J.F. Hughes, "Computer graphics: principles and practice", *Addison Wesley*, 1992.
- [14] <http://www.breuckmann.com/HTML/engl/tritos.html>, site web du capteur Tritos, société Breuckmann

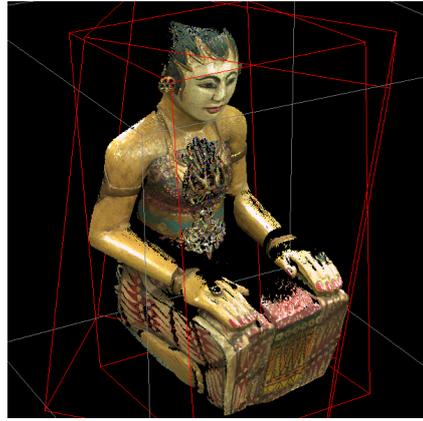


Figure 3 : Bali avant et après recalage

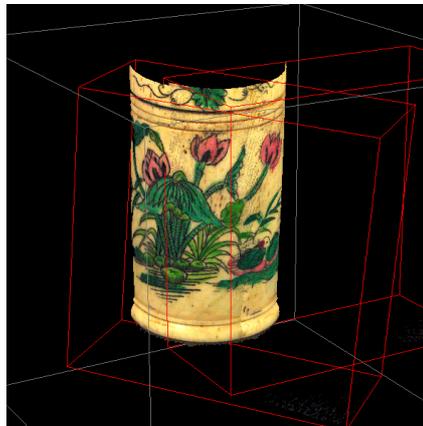
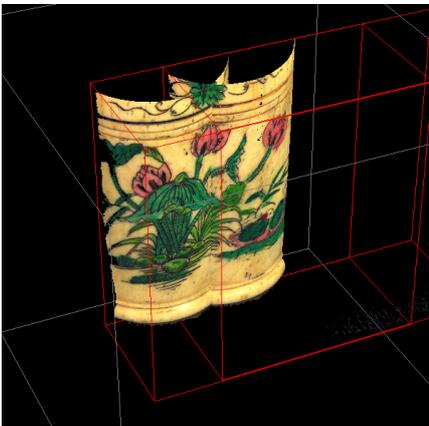


Figure 4 : Ivoire avant et après recalage

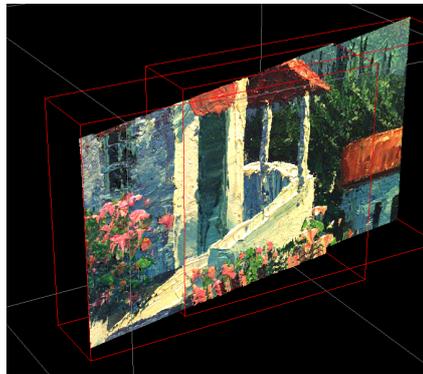
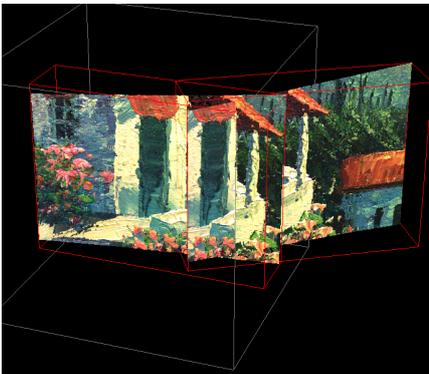


Figure 5 : Wallis avant et après recalage

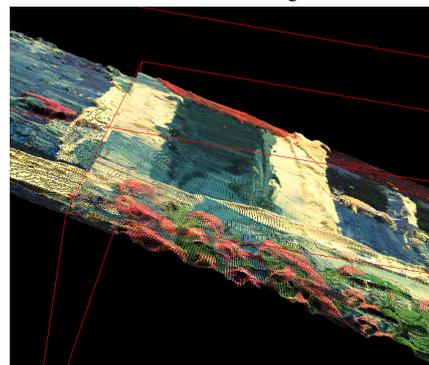


Figure 6 : Vue agrandie de la surface de Wallis après recalage