# Acquisition 3D des gestes par vision monoscopique en temps réel

**David Antonio Gómez Jáuregui — Patrick Horain — Fawaz Baroud**

*Institut TELECOM / TELECOM & Management SudParis / EPH*
*9 Rue Charles Fourier, 91011 EVRY Cedex, France*

*David.Gomez@IT-SudParis.eu,*

*Patrick.Horain@IT-SudParis.eu,*

*fbaroud@gmail.com*

RÉSUMÉ. *Nous nous intéressons à l'acquisition 3D des gestes humains par vision monoscopique en temps réel sans marqueur. Partant de travaux antérieurs sur le recalage par régions colorées d'un modèle 3D articulé du corps humain sur des séquences vidéo (Marques Soares et al., 2004), nous décrivons la mise en œuvre à la cadence vidéo d'un recalage par minimisation de la distance entre les contours de l'image et les contours occultants du modèle 3D extraits sur processeur graphique (GPU) grand public. Nous donnons quelques exemples de résultats et discutons le gain de précision obtenu.*

ABSTRACT. *Our topic is 3D human motion capture by real-time monocular vision without marker. We extend a previous work on colour-region based registration of a 3D articulated model of the human body on video sequences (Marques Soares et al., 2004). We describe a video frame-rate implementation for registering image edges with the 3D model occluding edges. The later are extracted using a consumer graphics processing unit (GPU).We give some example results and discuss the precision improvement achieved.*

MOTS-CLÉS : *Acquisition 3D des gestes, vision monoscopique, recalage 3D / 2D, contours occultants, recalage par carte de distance, calcul sur processeur graphique.*

KEYWORDS: *3D motion capture, monocular vision, 3D / 2D registration, occluding edges, chamfer matching, GPU computing.*

## 1. Introduction

Les gestes sont un moyen de communication entre personnes. Les environnements virtuels collaboratifs permettent une forme intuitive d'interaction gestuelle entre utilisateurs. Dans ces environnements, les utilisateurs peuvent être représentés par des avatars humanoïdes 3D, pour donner la sensation de présence. Ces humanoïdes peuvent être animés en utilisant des animations prédéfinies déclenchées par des commandes explicites. Toutefois, l'interaction est alors limitée à quelques animations prédéfinies. L'acquisition des gestes et leur restitution en temps réel par un avatar permet une interaction plus intuitive car les gestes de l'utilisateur peuvent être reproduits directement.

Dans ce travail, nous nous intéressons à l'acquisition 3D des gestes en temps réel par vision monoscopique et sans marqueur (Horain *et al.*, 2003), par exemple à partir d'une webcam ordinaire. Nous partons de travaux précédents dans lesquels l'acquisition des gestes est réalisée en recalant par régions colorées un modèle 3D articulé du corps humain sur des séquences vidéo. Dans cet article, nous proposons un recalage par minimisation de la distance entre les contours de l'image et les contours occultants du modèle 3D afin d'améliorer la précision de cette acquisition.

Nous présentons ci-après une brève description des travaux existants pour l'acquisition des gestes. Ensuite, nous présentons les travaux antérieurs de Marques Soares sur le recalage par régions colorées (Marques Soares, 2004), puis nous proposons une méthode pour intégrer les contours. Enfin, nous décrivons nos résultats expérimentaux, et nos conclusions.

## 2. Travaux existants pour l'acquisition 3D des gestes

L'intérêt de la recherche pour l'acquisition des gestes à partir d'images monoculaires est stimulé par de multiples applications potentielles (vidéosurveillance, interfaces homme machine, jeux, etc.) (Moeslund *et al.*, 2006). De nombreuses techniques ont été proposées pour l'initialisation automatique de la forme et de l'aspect du modèle ainsi que des détecteurs des parties du corps (Cheung *et al.*, 2003), l'utilisation de modèles de mouvement *a priori* (Urtasun *et al.*, 2004), l'échantillonnage stochastique (robuste aux occultations partielles) (Sminchisescu *et al.*, 2003), l'apprentissage de la correspondance entre l'image et la pose 3D (Agarwal *et al.*, 2006).

## 3. Notre approche pour l'acquisition des gestes

Notre méthode consiste à recaler de façon optimale la projection d'un modèle 3D articulé du corps humain sur chaque image de la séquence vidéo (Marques Soares *et al.*, 2004). Cette méthode est capable d'acquérir les gestes sans utilisation de modèles de gestes. Les paramètres du modèle pris en compte sont les 3 paramètres de position globale du corps ainsi que 20 angles des articulations de la partie supérieure du corps (buste, bras, avant-bras, mains, cou et tête). Ces paramètres sont issus de la norme H-ANIM (H-Anim, 2008). Chaque geste est ainsi représenté par un vecteur de paramètres. Des primitives (régions, contours) sont extraites des images vidéo d'une part et du modèle 3D d'autre part. Le recalage consiste à mettre en correspondance ces caractéristiques de façon optimale.

### 3.1. Recalage sur les régions

L'arrière-plan est grossièrement détecté par différence avec une image de référence. Les zones d'avant-plan sont ensuite segmentées en classes de couleur (peau, vêtement) dont la distribution est apprise à partir d'échantillons au début de la séquence vidéo. Un détecteur de visage Adaboost (Viola *et al.*, 2001), disponible dans la bibliothèque OpenCV d'Intel, permet de localiser le visage et d'obtenir un échantillon de la peau. Un échantillon du vêtement est extrait en dessous du visage. Ces échantillons sont ensuite modélisés par des distributions gaussiennes dans l'espace colorimétrique HSV (Bradski, 2002). La posture du modèle 3D est ajustée à l'image. Le modèle 3D placé dans la pose décrite par le vecteur de paramètres est ensuite projeté dans l'image. Les segments articulés ont des attributs de couleur (peau, vêtement) et se projettent comme des régions colorées. La correspondance entre la projection du modèle et l'image vidéo segmentée est évaluée par un taux de non recouvrement entre régions colorées :

$$F(q) = \prod_{c=1}^{m} \left( \frac{|\, A_c \cup B_c(q) \,| - |\, A_c \cap B_c(q) \,|}{|\, A_c \cup B_c(q) \,|} \right)^{\frac{1}{m}} \qquad [1]$$

où $q$ représente le vecteur des paramètres qui décrivent la posture candidate, $A_c$ est l'ensemble des pixels dans la $c^{\text{ème}}$ classe de couleur dans l'image vidéo segmentée, $B_c(q)$ est l'ensemble des pixels dans la $c^{\text{ème}}$ classe de couleur dans la projection du modèle, $m$ est le nombre de classes de couleur et $|\, X \,|$ représente le nombre de pixels dans $X$.

Cette fonction est ensuite minimisée par rapport à $q$. Toutefois, comme elle n'est pas facilement dérivable, l'utilisation d'un algorithme de descente de gradient n'est pas possible. L'algorithme de descente de simplexe (Nelder *et al.*, 1965) permet de minimiser itérativement ce taux, tout en respectant des contraintes biomécaniques (Marques Soares *et al.*, 2004). Il est important de remarquer que, pour converger

vers une position approximativement correcte, cette méthode ne nécessite qu'un recouvrement partiel entre régions colorées. Toutefois, elle peut ne pas être précise car les pixels de la frontière des régions sont peu nombreux par rapport aux pixels de l'intérieur de la région (figure 1).

**Figure 1.** *Précision limitée du recalage sur les régions : les images sont respectivement l'image acquise puis segmentée, la projection du modèle 3D et enfin la superposition de la projection du modèle avec l'image segmentée. La pose du modèle 3D diffère de celle de l'acteur observé car le recalage sur les régions n'est pas précis.*

### 3.2. Recalage sur les contours

Pour augmenter la précision du recalage, nous proposons de mettre en correspondance les contours de l'image avec les contours occultants du modèle en minimisant la distance qui les sépare (Lu *et al.*, 2002 ; Sminchisescu *et al.*, 2003). Ce recalage sur les contours réalise un appariement implicite entre les points de contours les plus proches qui, pour être correct, requiert que l'état initial du modèle soit proche de l'optimum recherché, sous peine que l'algorithme converge vers un optimum local. En contrepartie, l'ajustement des contours peut être plus précis sur les détails de l'objet. Nous avons donc fait suivre l'étape précédente de recalage sur les régions, relativement robuste mais imprécise, par une étape de recalage sur les contours destinée à améliorer la précision du recalage.

#### *3.2.1. Extraction des contours dans les images*

Les contours dans les images sont extraits par le filtre de Canny puis seuillés avec hystérésis (Horaud *et al.*, 1995). Une transformation distance des contours, qui affecte à chaque pixel sa distance au contour le plus proche (Borgefors, 1986), produit une carte de distance aux contours (figure 2).

**Figure 2.** *Exemple d'image issue de l'acquisition vidéo, les contours extraits de cette image et la carte de distance aux contours.*

*3.2.2. Détection des contours occultants du modèle*

Les contours occultants d'un objet 3D sont formés des points de la surface où la direction d'observation est tangente à la surface (Franco *et al.*, 2003). Le produit scalaire entre le vecteur normal à la surface et la direction d'observation change de signe de part et d'autre de ces lignes.

Ils peuvent être extraits simplement et efficacement en utilisant l'interface de programmation OpenGL. Cette opération par projection avec élimination des éléments de surface selon leur orientation vers l'avant ou vers l'arrière de l'observateur (*culling*) est simple à programmer en OpenGL et est rapide car elle peut être exécutée efficacement par le processeur graphique (GPU). Les polygones du maillage orientés vers l'observateur ou parallèles à la direction d'observation sont d'abord projetés et remplis avec leurs arêtes par une couleur différente du fond de l'image. Ensuite l'intérieur de ceux strictement orientés vers l'arrière de la scène est rempli avec la couleur de fond. Ainsi, seules les arêtes occultantes restent marquées différemment du fond (figure 3).

***Figure 3.*** *Les contours occultants du modèle 3D.*

*3.2.3. Recalage sur les contours*

La distance de chaque point de contour occultant au contour le plus proche dans l'image vidéo est lue directement dans la carte de distance précédemment calculée. La distance résiduelle entre contours occultants et contours de l'image vidéo est la moyenne de la carte de distance masquée par l'image binaire des contours occultants. Le recalage sur les contours consiste à minimiser cette distance entre contours par l'algorithme de descente de simplexe déjà utilisé à l'étape précédente.

## 4. Mise en oeuvre et résultats

La séquence vidéo au format 160 × 120 pixels est obtenue au moyen d'une webcam Logitech QuickCam Pro 5000. La segmentation en régions colorées, l'extraction des contours et la transformation distance sont mises en œuvre avec la bibliothèque OpenCV (Bradski, 2002). Les manipulations du modèle 3D sont effectuées par la bibliothèque OpenGL fournie avec le GPU.

Nous avons mis en œuvre le recalage sur les régions colorées suivi d'un recalage sur les contours. Nos résultats (figure 4) montrent que l'étape de recalage sur les contours permet pour certaines images de corriger des recalages sur les régions

incorrects qui se manifestent comme des pics de la distance résiduelle entre contours. Dans cette séquence vidéo, nous avons trouvé que les pics d'erreur qui peuvent être corrigés représentent le 25% des images capturées. La figure 5 permet d'illustrer un exemple d'un recalage corrigé en utilisant l'étape de recalage sur les contours.

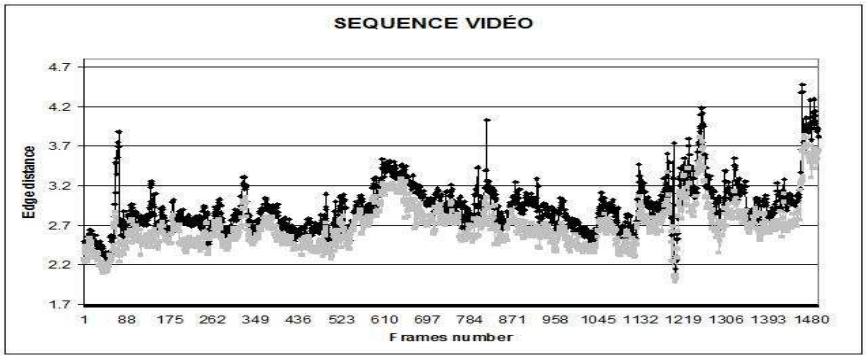


**Figure 4.** *Distance entre contours résiduelle avec le recalage sur les régions colorées (en noir) suivi du recalage sur les contours (en gris).*

**Figure 5.** *Exemple de la correction d'un recalage sur les régions incorrect. Les images sont respectivement l'image acquise puis la superposition de la projection du modèle avec l'image segmentée qui montre le recalage sur les régions incorrect, les contours occultants du modèle 3D qui montrent la correction du recalage sur les contours et enfin la projection de la pose du modèle 3D corrigée.*

Dans notre approche, le nombre d'itérations (et donc le temps de calcul) de l'algorithme d'optimisation varie pour chaque image capturée selon la proximité de la solution. Afin de respecter la cadence vidéo de 25 images par seconde en utilisant la méthode proposée, l'optimisation est limitée à un nombre maximal de 30 itérations sur notre PC équipé d'un processeur Intel Core 2 Duo 2 à 1,7 GHz et une carte graphique NVIDIA GeForce 8800 GTS. La figure 6 permet de visualiser les résultats de ces expérimentations réalisées. Le temps de calcul disponible pour chaque image doit être réparti entre les 2 étapes successives de recalage. Nous avons fait des expérimentations avec la même séquence vidéo en remplaçant progressivement les itérations de recalage sur les régions colorées par des itérations de recalage sur les contours, pour essayer de réduire l'erreur finale de recalage à temps de calcul constant.

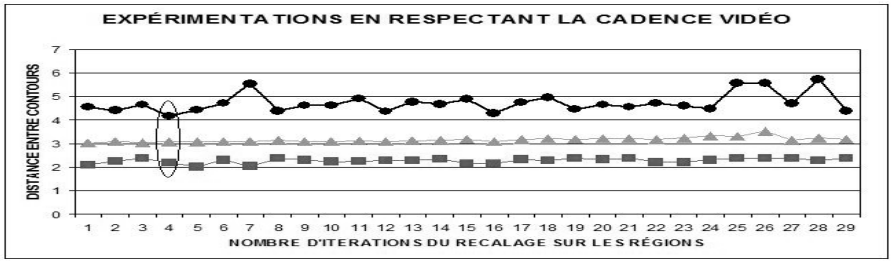


**Figure 6.** *Résultats des expérimentations réalisées en respectant la cadence vidéo. Dans l'axe des abscisses, chaque nombre d'itérations sur les contours est égal à 30 moins le nombre d'itérations sur les régions. Les cercles, triangles et carrés sont respectivement les valeurs maximales, les valeurs moyennes et les valeurs minimales de la distance entre contours résiduelle obtenues pour chaque expérimentation. Les valeurs entourées représentent les itérations optimales (les valeurs maximales, moyennes et minimales qui sont les plus proches)*

Par rapport aux résultats obtenus, nous avons observé qu'un nombre d'itérations de 4 pour le recalage sur les régions et 26 pour le recalage sur les contours permet de réduire mieux l'erreur du recalage en conservant la cadence vidéo. De cette manière, nous observons qu'il faut un plus grand nombre d'itérations pour le recalage sur les contours par rapport au nombre d'itérations du recalage sur les régions afin d'obtenir les meilleurs résultats en conservant le temps de calcul constant.

## 5. Conclusions et perspectives

Nous proposons une méthode pour l'acquisition des gestes par vision artificielle monoscopique combinant un recalage sur des régions colorées et sur les contours. Nos expérimentations montrent que l'utilisation d'un recalage sur les contours après d'un recalage sur les régions, permet d'améliorer la précision. Par conséquent, il est possible de déduire, que l'utilisation des contours permet à l'algorithme d'optimisation de trouver des meilleures solutions pour l'acquisition 3D des gestes. Afin de réduire ce temps de calcul par itération, et ainsi augmenter le nombre d'itérations par image tout en respectant la cadence vidéo, nous envisageons d'exploiter la puissance des cartes graphiques pour le traitement des images (Farrugia *et al.*, 2006). Nous envisageons aussi d'explorer et combiner d'autres types d'information comme le mouvement, les textures (Lu, 2002) ainsi que d'autres algorithmes d'optimisation afin d'augmenter la robustesse et la précision de notre approche.

## 6. Bibliographie


Agarwal A., Triggs B., « Recovering 3D human pose from monocular images », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, janvier 2006, p.44–58.

Borgefors G., « Distance transformations in digital images », *Computer Vision, Graphics and Image processing*, vol. 34, 1986, p.344-371.

Bradski, G., « OpenCV: Examples of Use and New Applications in Stereo, Recognition and Tracking », *15th International Conference on Vision Interface*, Calgary, Canada, 27-29 mai 2002.

Cheung G., Baker S., Kanade T., « Shape-from-silhouette for articulated objects and its use for human body kinematics estimation and motion capture », *Computer vision and pattern recognition*, Madison, Wisconsin, USA, 16-22 juin 2003.

Farrugia J.P., Horain P., Guehenneux E., Allusse Y., « GPUCV: A framework for image processing acceleration with graphics processors», *CDROM proc.of the IEEE International Conference on Multimedia & Expo ICME 2006*, Toronto, Ontario, Canada, 9-12 juillet 2006.

Franco J.-S., Boyer E., « Une approche hybride pour calculer l'enveloppe visuelle d'objets complexes », *ORASIS'03*, mai 2003, p. 67-74.

Horain P., Bomb M., « Acquisition du geste humain 3D par vision monoscopique », Actes des *8èmes journees d'études et d'échanges Compression et Représentation des Signaux Audiovisuels CORESA'03*, Lyon, 16-17 janvier 2003, p. 269-272.

Horaud R., Monga O., *Vision par ordinateur outils fondamentaux*, Hermès, 1995.

Humanoid Animation Working Group, *H−ANIM specification*. http://H-Anim.org., 2008

Lu S., Huang G., Samaras D., Metaxas D., « Model-based integration of visual cues for hand tracking », *Proceedings of IEEE workshop on Motion and Video Computing*, Orlando, Florida, 3-4 decembre 2002, p. 118-124.

Marques Soares J., Horain P., Bideau A., Nguyen M.H., « Acquisition 3D du geste par vision monoscopique en temps réel et téléprésence », Actes de l'atelier *Acquisition du geste humain par vision artificielle et applications*, Toulouse, 27 janvier 2004, p. 23-27.

Moeslund T., Hilton A., Kruger V., « A survey of advances in vision-based human motion capture and analysis », *Computer vision and image understanding*, vol. 4, novembre-décembre 2006, p. 90-126.

Nelder J. A., Mead R. « A simplex method for function minimization », *Computer Journal*, vol. 7, 1965, p. 208-313.

Sminchisescu C., Triggs B., « Estimating Articulated Human Motion with Covariance Scaled Sampling », International Journal of Robotics Research, vol. 22, n° 6, 2003, p. 371-393.

Urtasun R., Fua P. « 3D human body tracking using deterministic temporal motion models », *European Conference on Computer Vision*, Prague, Czech Republic, 11-14 mai 2004.

Viola P., Jones M., « Rapid Object Detection Using a Boosted Cascade of Simple Features », *IEEE Computer Vision and Pattern Recognition*, vol. 1, 2001, p. 511