

# Suivi 3D à partir d'un modèle basé points

Christophe Dehais<sup>1</sup>

Vincent Charvillat<sup>1</sup>

Géraldine Morin<sup>1</sup>

<sup>1</sup> IRIT - ENSEEIHT

Site ENSEEIHT

2, rue Charles Camichel - BP4122 - 31071 Toulouse Cedex 7

{dehais, charvi, morin}@enseeiht.fr

## Résumé

*Cet article présente une déclinaison originale d'un des meilleurs algorithmes de suivi 3D temps réel issu de l'état de l'art. Il s'agit d'un algorithme de suivi basé sur un modèle 3D par facettes planes de l'objet à suivre. Notre contribution essentielle est de substituer ce modèle basé sur des facettes par un nouveau modèle basé sur des points 3D. Cette proposition nous conduit à reformuler le problème en faisant apparaître plusieurs étapes d'estimation linéaire de paramètres. Ces linéarisations permettent une implantation moins complexe et moins coûteuse de l'algorithme initial. Des expériences montrent finalement que son efficacité est conservée.*

## Mots clefs

Suivi 3D, modèle basé points

## 1 Introduction

De nombreuses applications nécessitent le suivi 3D d'objets rigides (ou peu déformables). C'est le cas en particulier de la Réalité Augmentée [1], de la commande visuelle de robots [2] ou du suivi de visage [3].

Dans ces contextes, des contraintes de coût ou des difficultés d'instrumentation de l'environnement conduisent à la recherche de solutions exploitant la vision par ordinateur. Récemment les méthodes s'appuyant sur la connaissance d'un modèle 3D de l'objet d'intérêt et sur la détection et le suivi des caractéristiques naturelles de celui-ci (contours, points d'intérêts, textures) ont montré des résultats très satisfaisants [4, 5].

Les algorithmes proposés ont pour objectifs :

- l'efficacité algorithmique : les applications mentionnées plus haut doivent suivre l'objet à la cadence d'acquisition de la vidéo, entre 10 et 30 images/sec.
- la robustesse : dans la pratique, les données visuelles extraites des images et utilisées pour le suivi sont souvent imprécises et parfois aberrantes. Un algorithme de suivi robuste doit résister à ces contaminations.
- la précision : deux types de problèmes sont fréquemment rencontrés. L'accumulation d'erreurs au cours du suivi conduit d'une part à une dérive progressive du résultat

par rapport à la solution recherchée. D'autre part, l'instabilité des paramètres retrouvés induit une cohérence temporelle insuffisante. Ce dernier problème est particulièrement gênant dans le cas de la Réalité Augmentée.

Vacchetti et al. [3] proposent un algorithme temps réel qui remplit ces exigences de robustesse et de précision. Ils s'appuient sur la connaissance a priori d'un modèle composé de facettes (maillage) de l'objet à suivre. Ceci permet de simplifier l'expression d'un problème d'ajustement de faisceaux à partir de mises en correspondance de points d'intérêt. Une faible dérive des paramètres et une bonne stabilité temporelle sont obtenues par l'utilisation conjointe d'un suivi itératif et d'un suivi par rapport à une image clé. Bien qu'une implantation en temps réel de cette technique soit possible [3], elle reste délicate à obtenir pour plusieurs raisons. D'abord les critères à optimiser sont non-linéaires, et présentent de nombreux minima-locaux, ce qui nécessite une initialisation suffisamment proche de la solution. Seules des méthodes d'optimisation itératives relativement coûteuses sont utilisables. Ensuite l'environnement de mise en oeuvre fait appel à des calculs intermédiaires nombreux (en particulier pour associer des indices visuels 2D extraits des images aux "bonnes" facettes 3D du modèle). Ces difficultés d'implantations compromettent la viabilité de futures extensions de cette approche, en particulier pour traiter explicitement les déformations.

Par ailleurs, des modèles 3D qui ne sont plus nécessairement complets ou organisés topologiquement et qui incorporent des données textuelles en plus de la géométrie [6, 7] ont montré leur intérêt en vision :

- pour leur capacité en reconnaissance [8]
- pour le suivi avec des approches de type appariement de motif utilisant des modèles d'apparence. [6, 7]

Il faut remarquer que cette tendance est également observable en synthèse d'images où le rendu par points devient performant [9].

Suivant cet élan, nous proposons de bénéficier des avantages des modèles à base de points pour implanter une version simplifiée de l'algorithme présenté par Vacchetti et al. [3] (notre algorithme *de référence* pour cette étude).

Dans la section 2 nous présentons les éléments de modélisation du problème. Les détails algorithmiques de

notre approche sont présentés en section 3. Des expériences sur des données synthétiques et réelles sont enfin reportées en section 4. Nous comparons en particulier nos algorithmes à ceux de notre implantation de l'algorithme de référence.

## 2 Suivi d'un modèle 3D par facettes

### 2.1 Modélisation, résolution du problème

On définit l'objectif du suivi comme l'estimation des paramètres de la pose d'un objet par rapport à la caméra au cours d'une séquence d'image  $\{I_t\}$ .

**Le modèle de caméra.** La caméra perspective est modélisée par une matrice de projection  $P$   $3 \times 4$ , qui peut se décomposer sous la forme :

$$P = K.[R | \mathbf{t}]$$

où  $K$  est la matrice triangulaire supérieure comprenant des paramètres *intrinsèques* de la caméra (distance focale, géométrie du capteur). Nous supposons que  $K$  est connue et constante au cours de la séquence.  $R$  et  $\mathbf{t}$  représentent respectivement la matrice de rotation et le vecteur de translation définissant la transformation d'un repère lié à l'objet au repère de la caméra.  $R$  et  $\mathbf{t}$  varient aux cours du temps. On notera donc  $P_t = K.[R_t | \mathbf{t}_t]$  la pose de l'objet à l'instant  $t$  (associée à l'image  $I_t$  de la séquence).

L'objectif est de retrouver pour chaque image les paramètres de rotation et de translation, soit 6 paramètres.

**Mesures dans les images.** Plusieurs types d'indices visuels et méthodes peuvent être utilisés pour retrouver le mouvement 3D à partir de déplacements 2D : contours, motifs texturés, flot optique. Les méthodes présentées plus bas s'appuient sur l'extraction et la mise en correspondance de points d'intérêt. Cette approche présente plusieurs avantages : elle peut-être rendue robuste aux changements d'illumination, aux occultations locales, et elle supporte des mouvements plus large que les méthodes de flot optique ou de suivi de motifs.

Des points d'intérêt selon le critère de Harris [10] sont mis en correspondance entre deux images consécutives  $I_{t-1}$  et  $I_t$ . La mesure de similarité utilisée est une corrélation croisée normalisée de fenêtres centrées en les points candidats. Elle est invariante aux changements affines d'illumination.

Soient  $p_{t-1}^j$  un point d'intérêt dans l'image  $I_{t-1}$  et  $p_t^i = p_t^{v(j)}$  le point apparié dans l'image  $I_t$  (voir Figure 1). Si  $p_{t-1}^j$  appartient à l'image de l'objet, il est la projection selon  $P_{t-1}$  d'un point 3D  $N^j$  appartenant à la surface définie par le modèle.  $N^j$  se projète dans  $I_t$  en le point  $\tilde{p}_t^j$ .

**Résolution.** On recherche donc  $P_t = K.[R_t | \mathbf{t}_t]$  qui minimise conjointement les distances  $\|\tilde{p}_t^j - p_t^{v(j)}\|^2$  pour toutes les correspondances de points.

Il s'agit en fait d'un problème particulier d'ajustements de faisceaux, mais ici la connaissance des facettes permet de s'affranchir de la détermination explicite des points 3D  $N^j$ .

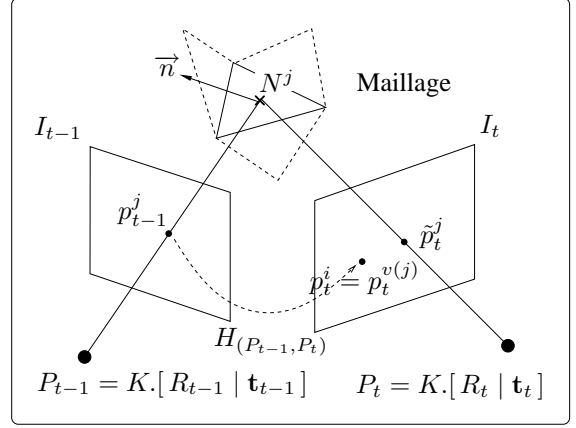


Figure 1 – Représentation géométrique du critère de (1)

En effet,  $p_{t-1}^j$  et  $\tilde{p}_t^j$  sont reliés par une homographie qui ne dépend que des poses  $P_{t-1}$  et  $P_t$  et de la facette  $\mathcal{F}$  à laquelle appartient  $N^j$  [11]. Ainsi si  $\{x | \mathbf{n}^T \cdot x - d = 0\}$  est le plan de la facette  $\mathcal{F}$  ( $\mathbf{n}$  est sa normale), alors on a :

$$\tilde{p}_t^j = H_{(\mathcal{F}, P_{t-1}, P_t)} \cdot p_{t-1}^j$$

où

$$H_{(\mathcal{F}, P_{t-1}, P_t)} = K.(\delta R - \delta \mathbf{t} \cdot \mathbf{n}'^T / d').K^{-1}$$

$$\delta R = R_t \cdot R_{t-1}^T, \quad \delta \mathbf{t} = -\delta R \cdot \mathbf{t}_t + \mathbf{t}_{t-1}$$

$$\mathbf{n}' = R_{t-1} \cdot \mathbf{n}, \quad d' = d - \mathbf{t}_{t-1}^T \cdot (R_{t-1} \cdot \mathbf{n})$$

A chaque image, on est donc amené à résoudre le problème d'optimisation (sur les 6 paramètres de  $P_t$ ) *non-linéaire* suivant :

$$\hat{P}_t = \operatorname{argmin}_{P_t} \sum_j \|H_{(\mathcal{F}(p_{t-1}^j), P_{t-1}, P_t)} \cdot p_{t-1}^j - p_t^{v(j)}\|^2 \quad (1)$$

En pratique, les erreurs de localisation sur les points d'intérêt portant à la fois sur  $p_{t-1}^j$  et sur  $p_t^i$ , on symétrise le problème en transportant de la même manière  $p_t^i$  dans l'image  $I_{t-1}$ . Le critère est la somme de 2 distances, et à chaque étape l'estimation porte donc sur les 12 paramètres variables de  $P_{t-1}$  et  $P_t$ .

Ce problème aux moindres carrés non-linéaires est résolu itérativement par la méthode de Levenberg-Marquadt. Le suivi itératif ainsi posé est intéressant car il n'utilise aucune connaissance a priori sur la séquence (sauf la pose initiale). Cependant il souffre de l'accumulation des erreurs d'estimation, ce qui empêche toute utilisation sur des séquences plus longues que quelques centaines d'images. Pour pallier à ce problème [3] propose d'exploiter des informations a priori, constituées hors-ligne sous la forme d'images clés.

### 2.2 Intégration d'informations a priori

Une image clé est constituée d'une vue de l'objet à suivre et d'un recalage manuel précis du modèle. Les points

d'intérêts  $p_{cle}^j$  détectés dans ces images ainsi que leurs antécédents 3D  $N_{cle}^j$  sont précalculés.

Pour chaque pairs de points appariés  $(p_t^j, p_{cle}^{w(j)})$  entre l'image courante  $I_t$  et l'image clé  $I_{cle}$ , on peut compléter (1) en sommant également sur les termes :

$$\|p_t^j - \phi(P_t, N_{cle}^{w(j)})\|$$

où  $N_{cle}^{w(j)}$  est l'antécédent 3D de  $p_{cle}^{w(j)}$  et  $\phi(P, \cdot)$  est l'opérateur de projection selon la matrice  $P$ .

Il y a deux difficultés : la première est la mise en correspondance des points entre la vue courante et une image clé, qui peuvent être relativement éloignées. Vacchetti et al. [3] résolvent cela en synthétisant une version déformée de l'image clé dans laquelle les mesures de corrélation sont valides. La deuxième difficulté est le choix de l'image clé. On peut choisir celle dont les paramètres de pose sont les plus proches de la pose connue la plus récente  $P_{t-1}$ , mais ce choix est parfois sous-optimal pour l'objectif de mise en correspondance. Il est préférable d'utiliser une mesure de similarité entre l'image courante et l'image clé candidate, comme proposé dans [3].

### 3 Notre approche avec un modèle par points

La solution présentée à la section 2 est intéressante mais la source de ses difficultés d'implantation repose sur l'utilisation de facettes pour définir un modèle de mouvement 2D (homographique) utilisé sur un ensemble peu dense de points (les points d'intérêts). L'idée que nous proposons d'explorer est de rendre homogène l'approche de suivi 2D avec la modélisation de l'objet. Ainsi nous prétendons pouvoir nous passer des facettes, et ainsi gagner en souplesse (voir section 5). Dans la section suivante nous présentons le modèle à base de points que nous utilisons par la suite.

#### 3.1 Le modèle

Le modèle utilisé est composé d'un ensemble de *patches* définis par un centre  $M_i = (x_i, y_i, z_i)^T$ , et une normale  $\mathbf{n}$  (voir Figure 2). Il s'agit d'un modèle similaire à celui utilisé par Muñoz et al. [7].

#### 3.2 Un modèle de mouvement adapté aux points

Le mouvement 3D entre 2 images consécutives  $I_{t-1}$  et  $I_t$  est décrit par une transformation euclidienne 3D de matrice  $4 \times 4$  :

$$M = \begin{bmatrix} \delta R & \delta \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}$$

où  $\delta R$  et  $\delta \mathbf{t}$  représentent la rotation et la translation subit par l'objet entre les poses  $P_{t-1}$  et  $P_t$ . Ainsi :

$$P_t = P_{t-1} \cdot M$$

Pour obtenir un modèle de mouvement 2D adapté à l'utilisation de points, nous linéarisons ce modèle de mouvement à la manière de Drummond [4].

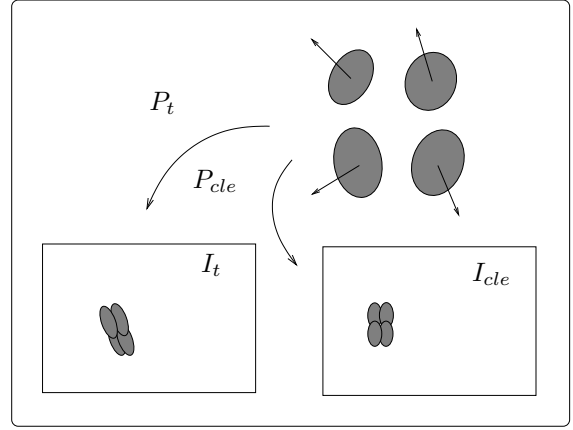


Figure 2 – Le modèle défini par ensemble de patches texturés, projeté selon une pose courante ou selon la pose d'une image clé (voir section 3.5).

$M$  peut s'écrire sous forme exponentielle :

$$M = \exp\left(\sum_{i=1}^6 \alpha_i G_i\right)$$

où les matrices  $G_i$  forment une famille génératrice des mouvements 3D élémentaires (rotations et translations par rapport aux axes du repère de l'objet) :

$$G_i = \begin{bmatrix} 0 & -\delta_{i,6} & \delta_{i,5} & \delta_{i,1} \\ \delta_{i,6} & 0 & -\delta_{i,4} & \delta_{i,2} \\ -\delta_{i,5} & \delta_{i,4} & 0 & \delta_{i,3} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \delta_{i,j} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases}$$

et  $\alpha = (\alpha_1, \dots, \alpha_6)$  sont les paramètres de translations et de rotations correspondants.

Sous l'approximation :

$$M \approx I + \alpha_i \cdot G_i \quad (2)$$

on peut calculer une famille génératrice  $\{L_{j,i}\}_{i=1,\dots,6}$  de vecteurs de mouvement 2D en tout point  $p^j = (u, v, w)$  projection dans le plan image d'un point 3D  $N^j = (x, y, z, 1)^T$ . On a :

$$\forall i \in [1, 6], \mathbf{L}_{j,i} = \begin{pmatrix} \frac{u'w - uw'}{w^2} \\ \frac{v'w - vw'}{w^2} \end{pmatrix} \text{ avec } \begin{pmatrix} u' \\ v' \\ w' \end{pmatrix} = P \cdot G_i \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

La famille  $\{L_{j,i}\}$  permet d'exprimer le mouvement 2D induit par un mouvement 3D paramétré par  $\alpha$ . La Figure 3 montre le champ des vecteurs  $L_{j,6}$  correspondant à une rotation autour de l'axe  $z$ .

Au contraire, notre approche a pour objectif de retrouver le mouvement 3D à partir de nombreuses mesures du mouvement local 2D dans les images. Cette approche a été précédemment utilisée avec des mesures échantillonnées le

long des arêtes d'un modèle [4, 3, 2], à notre connaissance jamais dans le cas de points isolés.

La Figure 4 illustre ce modèle de mouvement. Pour le calcul de la famille  $\{\mathbf{L}_{j,i}\}$  correspondant à un point quelconque  $p_{t-1}^j$ , voir la section 3.4.

Dans ce schéma, le suivi revient à estimer la matrice de la transformation euclidienne inter-image  $M$ , via ses paramètres  $\alpha$ . La section suivante montre comment obtenir très simplement ce résultat.

### 3.3 Résolution

Chaque appariement fournit une évaluation d'un vecteur de mouvement  $\mathbf{d}_j$  :

$$\mathbf{d}^j = p_{t-1}^j - p_t^{v(j)}$$

Pour une transformation euclidienne  $M$  telle que  $P_t = P_{t-1}.M$ , sous l'approximation de l'équation (2), on a :

$$\mathbf{d}^j = \sum_{i=1}^6 \alpha_i L_{j,i}$$

Soit  $\mathbf{d}^j = (d_x^j, d_y^j)^T$  le vecteur de mouvement,  $\mathbf{L}_{j,i}$  le  $i^{\text{ème}}$  vecteur de la famille génératrice calculés en  $p_{t-1}^j$  et  $L_j = (\mathbf{L}_{j,1}, \dots, \mathbf{L}_{j,6})$  la matrice  $2 \times 6$  contenant les 6 vecteurs de  $\{\mathbf{L}_{j,i}\}$ . On a la relation suivante :

$$D = L.\alpha = L. \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_6 \end{pmatrix} \quad (3)$$

où

$$D = \begin{pmatrix} d_1^x & d_1^y \\ \vdots & \vdots \\ d_J^x & d_J^y \end{pmatrix} \text{ et } L = \begin{pmatrix} \underline{L}_1 \\ \vdots \\ \underline{L}_J \end{pmatrix}$$

$J$  est le nombre d'appariements.

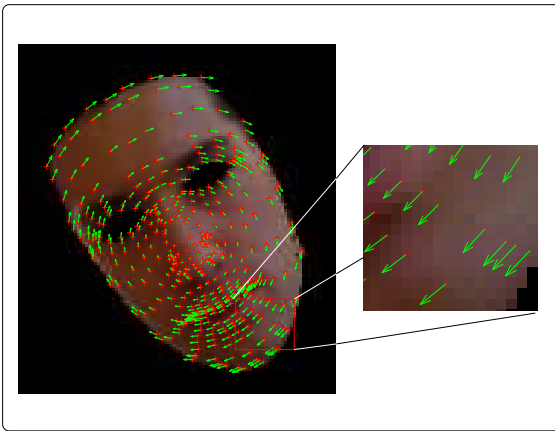


Figure 3 – Le champ des vecteurs  $\mathbf{L}_{j,6}$  correspondant à une rotation autour de l'axe  $z$ .

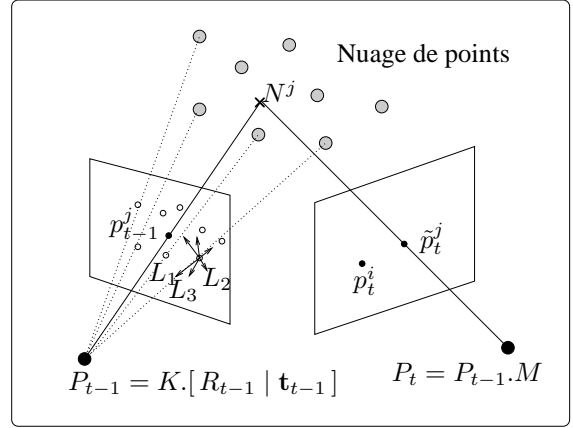


Figure 4 – Projection des générateurs  $G_i$  dans l'image. On a représenté la famille  $\{\mathbf{L}_{j,i}\}$  en la projection d'un point particulier du modèle

L'équation 3 lie de manière linéaire les paramètres du mouvement euclidien 3D et le mouvement apparent  $D$  calculé en les points d'intérêts par l'intermédiaire de la matrice  $L$ . La résolution de ce problème revient à évaluer la pseudo-inverse de  $L$  :

$$\alpha = L^+.D$$

Nous détaillons dans la section suivante les aspects algorithmiques supplémentaires intervenant dans la mise en oeuvre de ce schéma d'estimation.

### 3.4 Calcul du modèle de mouvement 2D

Nous mesurons les vecteurs de mouvement de manière éparse en des points d'intérêt, que nous considérons comme la projection d'un échantillonnage particulier de la surface définie par le nuage de points du modèle. Pour appliquer le schéma proposé plus haut, nous avons besoin de la connaissance de la famille  $\{\mathbf{L}_{j,i}\}$  en chacun des points d'intérêts  $p_j$ .

Nous proposons deux approches. La première utilise une projection arrière des points d'intérêts sur le modèle. Pour cela, nous exploitons une technique issue de la synthèse d'images qui consiste à calculer l'intersection du rayon issu du point d'intérêt avec une reconstruction locale de type MLS de la surface définie par le nuage [12]. Le coût de cette technique reste raisonnable pour un nombre de rayons limité (autant que de points d'intérêts appariés).

La deuxième possibilité consiste à considérer un maillage 2D dont les noeuds sont la projection des points du modèle. Le zoom de la Figure 3 montre qu'il est raisonnable d'interpoler localement le champ de mouvement calculé en les noeuds pour le reconstruire en n'importe quel point. Un schéma d'interpolation utilisant un modèle de type Thin Plate Spline [13] est utilisé pour évaluer les familles  $\{\mathbf{L}_{j,i}\}$  en tout point.

### 3.5 Intégration des connaissances a priori

La mise à jour itérative de la pose souffre de l'accumulation d'erreur d'estimation, ce qui peut conduire à l'échec du processus après quelques dizaines d'images (voir les résultats expérimentaux en section 4). Comme en section 2.2, nous introduisons donc des connaissances a priori sur la séquence grâce à des images clés, pour lesquelles la pose de l'objet est déterminée manuellement. Les points d'intérêts ainsi que leurs antécédents 3D sont également calculés. Nous procédons en deux étapes.

**Etape 1.** Il s'agit du schéma décrit en section 3.3. :

- extraction des points d'intérêts  $p_t^j$ ,
- formation de  $D^1$  (Eq 3) à partir des appariements  $(p_{t-1}^j, p_t^{v(j)})$ ,
- calcul des vecteurs  $L_{j,\cdot}^1$  en les points  $p_{t-1}^j$ , formation de la matrice  $L^1$ ,
- estimation de la mise jour  $\hat{\alpha}^1 = L^{1+} \cdot D^1$  des paramètres du mouvement 3D.

**Etape 2.** Raffinage de l'estimation courante grâce à une image clé.

- sélection de l'image clé et rendu des *patches* au voisinage de la pose courante (grâce à l'estimation de la pose courante  $P_t$ , disponible en sortie de l'Etape 1, cf. Figure 2),
  - formation de  $D^2$  à partir des appariements  $(p_t^j, p_{cle}^{w(j)})$ ,
  - calcul des vecteurs  $L_{j,\cdot}^2$  en les points  $p_t^j$ , formation de la matrice  $L^2$ ,
  - estimation raffinée grâce à la correction  $\hat{\alpha}^2 = L^{2+} \cdot D^2$ .
- Il faut noter que l'étape 2 est itérable.

## 4 Expériences

### 4.1 Données synthétiques

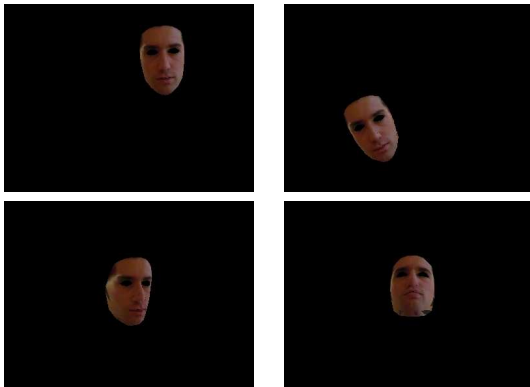


Figure 5 – 4 images issues de la séquence de synthèse.

Nous possédons un modèle CAO réaliste de visage, établi par un graphiste à partir de photographies. Il vient sous la forme d'un maillage composé de facettes triangulaires. Nous avons recalé manuellement ce modèle sur une vue réelle de visage pour obtenir des coordonnées de texture

en chaque point du maillage. A partir de ces données, nous avons produit un modèle par points tel que décrit en section 3.1. Celui-ci contient environ 500 points. Ces données nous ont permis de générer une séquence de synthèse par rendu OpenGL du maillage texturé. La séquence est obtenue en faisant évoluer les 6 paramètres de pose de la caméra entre chaque image de la séquence. Les paramètres utilisés pour générer la séquence constituent alors une vérité terrain à laquelle nous confrontons les valeurs estimées par les différentes méthodes implantées pour en évaluer la qualité. Toutes les méthodes évaluées ont été implantées en Matlab. La Figure 5 montre 4 images de la séquences de synthèse utilisée.

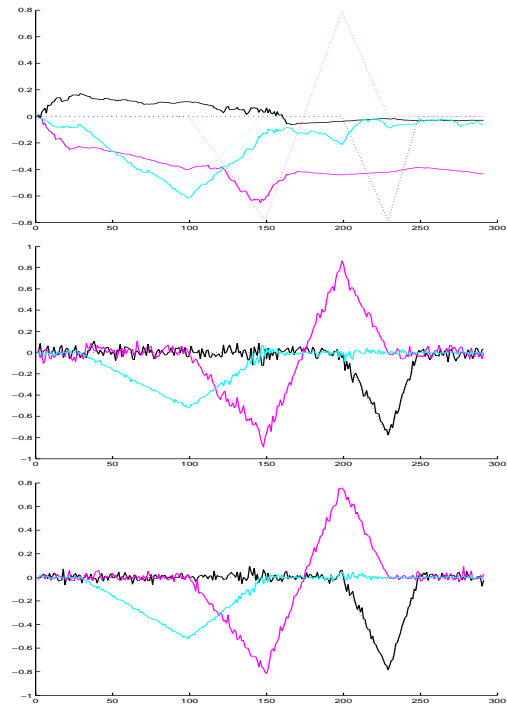


Figure 6 – Evolution des 3 paramètres de rotation estimés pour les 300 images de la séquence. Les courbes en pointillés correspondent à la vérité terrain. En haut : notre méthode sans image clé ; au milieu : notre méthode avec images clés ; en bas : la méthode non-linéaire de référence, avec images clés.

La Figure 6 présente les résultats du suivi sur une séquence de synthèse de 300 images exhibant différents mouvements de rotation et de translation du modèle devant une caméra perspective dont les paramètres sont fixés. L'erreur de localisation sur les points d'intérêt est gaussienne sans biais d'écart-type 2 pixels. Les courbes en pointillés représentent la vérité terrain telle qu'utilisée pour générer la séquence, les traits pleins sont les valeurs des paramètres estimés pour chaque méthode.

Deux conclusions peuvent être tirées de ces résultats : tout d'abord, que se soit dans la méthode de référence ou dans notre approche linéaire, l'utilisation des images clés corrige efficacement l'accumulation d'erreur.

Ensuite, malgré l'approximation que nous effectuons, notre méthode reste suffisamment stable bien que logiquement moins précise. Les composantes de  $\alpha$  étant du même ordre de grandeur, nous pouvons comparer l'erreur médiane absolue maximale; elle de l'ordre de  $10^{-2}$  pour notre méthode contre  $10^{-4}$  pour la méthode de référence. Ceci est obtenu pour un coût calculatoire bien plus faible. Ceci s'explique par le fait que notre approche, en linéarisant le problème au plus tôt, ne requière que l'évaluation de la pseudo-inverse de la matrice  $L$ , alors que le critère non-linéaire est évalué plusieurs centaines de fois (lors du calcul des jacobiniennes) dans la méthode non-linéaire.

## 4.2 Données réelles

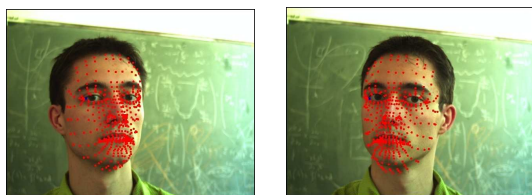


Figure 7 – Deux images de suivi d'une séquence réelle.

Nous avons également des résultats portant sur des séquences vidéo réelles acquises par une caméra couleur AVT Marlin F-046C calibrée (voir Figure 7). Le suivi de telles séquences nécessite de rendre robuste les différentes estimations intervenant dans notre algorithme. Nous utilisons une technique classique de moindres carrés pondérés itérés. Ceci consiste à pondérer les mesures issues de l'appariement des points d'intérêts apparaissant dans les lignes des matrices  $D$  (cf Eq. 3)

## 5 Conclusion et extensions envisagées

Nous avons démontré la possibilité d'utiliser un modèle s'appuyant uniquement sur des points pour le suivi visuel 3D. Cette technique est à la fois plus simple à mettre en oeuvre et plus rapide. De plus, elle apporte de la cohérence entre le modèle de l'objet et le traitement bas niveau (extraction de points d'intérêt).

Nous pensons également que l'utilisation d'un modèle sans topologie sera bénéfique dans la prise en compte d'objets déformables et pour le raffinement du modèle au cours du suivi. Voilà vers quoi nous souhaitons orienter nos futurs travaux.

## Références

[1] Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, et Blair MacIntyre. Recent advances in augmented reality. *IEEE Computer Graphics and Applications*, 21(6) :34–47, 2001.

[2] Andrew I. Comport, Éric Marchand, et François Chaumette. Robust model-based tracking for robot

vision. Dans *IEEE Int. Conf on Intelligent Robots and Systems, IROS04*, Sendai, Japan, September 2004.

- [3] Luca Vacchetti, Vincent Lepetit, et Pascal Fua. Stable real-time 3d tracking using online and offline information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10) :1391–1391, 2004.
- [4] Tom Drummond et Roberto Cipolla. Real-time visual tracking of complex structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7) :932–946, July 2002.
- [5] Luca Vacchetti, Vincent Lepetit, et Pascal Fua. Combining edge and texture information for real-time accurate 3d camera tracking. Dans *International Symposium on Mixed and Augmented Reality*, Arlington, VA, November 2004.
- [6] Charles S. Wiles, Atsuto Maki, et Natsuko Matsuda. Hyperpatches for 3d model acquisition and tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(12) :1391–1403, 2001.
- [7] Enrique Munoz, Jose M. Buenaposada, et Luis Baumela. Efficient model-based 3d tracking of deformable objects. Dans *Proceedings of ICCV 2005*, pages 877–882, Beijing, China, October 2005.
- [8] Frederick Rothganger, Svetlana Lazebnik, Cordelia Schmid, et Jean Ponce. 3d object modeling and recognition using affine-invariant patches and multi-view spatial constraints. Dans *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 272–277, Madison, WI, June 2003.
- [9] Mark Pauly. *Point Primitives for Interactive Modeling and Processing of 3D Geometry*. Thèse de doctorat, Federal Institute of Technology (ETH) of Zurich, 2003.
- [10] Chris Harris et Mike Stephens. A combined corner and edge detector. Dans *Fourth Alvey Vision Conference*, Manchester, 1988.
- [11] Richard Hartley et Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN : 0521540518, second édition, March 2004.
- [12] Anders Adamson et Marc Alexa. Ray tracing point set surfaces, 2003.
- [13] Fred L. Bookstein. Principal warps : Thin-plate splines and the decomposition of deformations. *IEEE Transaction on PAMI*, 11(6), 1989.