HAL
open science

Un algorithme efficace de suivi d'objets dans des séquences d'images

Frédéric Jurie, Michel Dhome

To cite this version:

Frédéric Jurie, Michel Dhome. Un algorithme efficace de suivi d'objets dans des séquences d'images. Reconnaissance des Formes et Intelligence Artificielle (RFIA '00), Feb 2000, Paris, France. pp.537–546. inria-00548292

HAL Id: inria-00548292

https://inria.hal.science/inria-00548292

Submitted on 21 Dec 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Un algorithme efficace de suivi d'objets dans des séquences d'images.

Tracking objects in image sequences : an efficient approach.

Frédéric Jurie Michel Dhome

LASMEA - CNRS UMR 6602, Université Blaise-Pascal, F-63177 Aubière
email : {Frederic.Jurie,Michel.Dhome}@lasmea.univ-bpclermont.fr

Résumé

Nous proposons dans cet article une approche permettant de suivre efficacement et rapidement le déplacement d'un motif visuel dans une séquence d'images. Cette technique consiste d'une part, en une étape hors ligne dédiée à l'apprentissage d'une matrice d'interaction liant la déformation du motif à son déplacement dans l'image, et d'autre part à l'exploitation en ligne de cette matrice pour suivre l'évolution du motif choisi. Cette seconde étape itérative consiste à prédire la position dans l'image du motif (en position, échelle et orientation), à calculer la différence du motif observé à l'endroit prédit avec le motif de référence, puis à réaliser le produit entre la matrice d'interaction et cette différence pour obtenir un vecteur correctif sur la position prédite. Nous montrons que cette étape de correction correspond à un coût algorithmique très faible permettant une mise en oeuvre en temps réel vidéo.
Dans la partie expérimentale, nous appliquons successivement ce principe au suivi d'un motif texturé dans une séquence d'images, puis au suivi d'objets volumiques (dans ce cas le motif de référence évolue dans le temps en fonction de l'orientation relative objet/caméra). De nombreux résultats expérimentaux sont présentés et commentés.

Mots Clefs

Vision par Ordinateur, Suivi d'Objets.

Abstract

This article describes an approach allowing to track efficently 2D patterns in image sequences. This approach consists of 2 stages. An off line stage is devoted to the computation of an interaction matrix linking the deformation of the shape with its displacement in the image. In the second stage this matrix is used to track the pattern : the position of the pattern is first predicted (the position, scale and orientation are predicted) ; then, the difference between the pattern observed at the predicted position and the pattern to track is computed. This difference multiplied by the interaction matrix gives a correction to apply to the prediction. We show that this correction stage has a very low algorithmic cost, allowing a real time implantation of the algorithm. We successively apply this framework to two applications : the tracking of 2D patterns and the tracking of 3D objects (in this case, the pattern evolves as the object is rotated). Numerous experimental results are given.

FIG. 1 – *Exemple d'objets que l'on souhaite pouvoir suivre dans une séquence d'images.*

Keywords

Computer Vision, Tracking.

1 Introduction

Nous nous intéressons, dans le cadre de cette étude, au problème du suivi d'objets mobiles dans des séquences d'images. Les objets d'intérêt sont de formes et de texture complexes (voir Figure 1). Un des domaines applicatifs visés est celui de la robotique manufacturière, où un bras manipulateur – muni d'une caméra embarquée – doit naviguer autour de ces objets et optimiser la trajectoire d'approche permettant de réaliser leur saisie.
Les algorithmes de suivi classiquement proposés dans la littérature reposent sur les deux étapes distinctes :

1. une phase de prédiction, durant laquelle une ou plusieurs hypothèses sur la position de l'objet dans l'image ou sur sa localisation par rapport à la caméra sont formulées,

2. une phase d'exploration d'un voisinage encadrant la prédiction, durant laquelle la position de l'objet

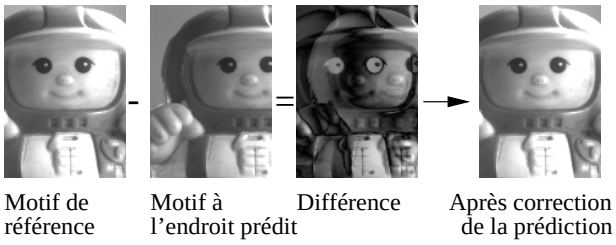

FIG. 2 – *Principe du suivi: le calcul de la différence entre le motif prédit et le motif de référence permet de corriger la position prédite.*

dans l'image ou l'attitude précise de l'objet est optimisée.

Nous montrons dans cet article, et c'est là que réside l'originalité principale de l'approche proposée, qu'il est possible, pour un domaine applicatif donné, de supprimer cette seconde phase. Cela est d'autant plus intéressant qu'elle s'avère généralement la plus coûteuse en terme de coût algorithmique.
La méthode que nous proposons consiste à mesurer l'erreur entre le motif de référence à suivre et le motif observé à l'endroit prédit, et à exploiter cette différence pour corriger les erreurs entachant la prédiction. Ce principe est illustré Figure 2. L'étape de correction nécessite une phase d'apprentissage hors ligne durant laquelle on estime la relation liant d'une part la différence entre la position prédite et la position réelle et d'autre part les modifications de l'aspect du motif. En pratique, le résultat de cette phase d'apprentissage est une matrice dite d'*interaction* qui, multipliée par la différence entre le motif courant et le motif de référence donne la correction à apporter sur les paramètres caractérisant la localisation prédite.
Les méthodes de suivi proposées dans la littérature portent soit sur le suivi de l'objet dans l'image, soit sur le suivi de l'attitude 2D ou 3D de l'objet.
Dans le premier cas le motif est suivi globalement, la recherche se faisant par exemple en maximisant un critère de corrélation entre un vecteur de luminance caractérisant le motif de référence et le contenu de l'image analysée [4, 2]. Les temps de traitement, importants dans ce cas, peuvent être réduits en travaillant dans des espaces de dimension plus faible que celui de représentation initiale de l'image [1, 10].
Dans l'autre cas, ce sont des primitives visuelles qui sont détectées et suivies dans les images [7]. Il peut s'agir de points d'intérêts [5], de segments [3], de contours [6], ou encore de régions [11]. Cela permet de localiser l'objet [8] dans l'image courante et de prédire la position des indices dans l'image suivante, en accord avec un modèle de mouvement de l'objet [12, 13] et un modèle de l'incertitude [9].
Bien que le suivi global du motif soit recherché dans le cadre de cette étude, la méthode proposée pourrait s'adapter au suivi de primitives en l'appliquant séparément à chacune d'entre elles.

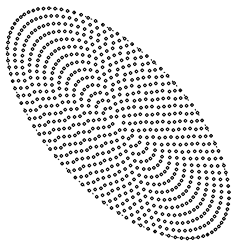

FIG. 3 – *Échantillonnage à l'intérieur d'une zone elliptique. Les points de la figure représentent les points où sont mémorisés les valeurs des niveaux de gris de l'image.*

L'article se décompose en quatre parties. Dans un premier temps, nous voyons comment le motif à suivre est caractérisé de façon à rendre sa représentation la plus indépendante possible de la position, de l'orientation et de l'échelle du motif dans l'image. Ensuite nous introduisons la notion de matrice d'interaction, nous montrons comment la calculer et l'exploiter; puis nous évaluons les performances de l'approche proposée par le biais d'exemples concrets. Dans la troisième partie de cet article, nous nous attachons à montrer comment les principes énoncés peuvent s'appliquer au suivi temps-réel de motifs visuels dans les images. L'étude du suivi d'objets volumiques fait l'objet de la dernière partie de cet article. Il s'agit alors de suivre un motif dans l'image, mais également d'autoriser et de gérer son évolution dans le temps en fonction de l'orientation relative caméra/objet.

2 Invariance de la représentation du motif

Nous souhaitons représenter le motif à suivre par un *vecteur de forme* quasi-indépendant de sa position, de son orientation et de son échelle dans l'image. Pour cela nous proposons d'échantillonner l'image à l'intérieur d'une zone elliptique, comme représenté Figure 3. Les points où sont réalisés les échantillonnages (niveaux de gris interpolés linéairement) sont répartis sur un ensemble d'ellipses concentriques déduites par homothétie de l'ellipse englobante. Ces ellipses sont échantillonnées de la plus petite à la plus grande; le nombre de points sur chacune d'elles étant prédéfini pour représenter un pas de parcours quasi constant. L'échantillonnage débute à partir de l'orientation du demi grand axe supérieur de l'ellipse.
L'ensemble des valeurs échantillonnées est ainsi stocké toujours dans le même ordre, dans ce que nous appelons un *vecteur de forme*. Ainsi, quelque soit la position, l'orientation et la taille d'un motif, sa repré-

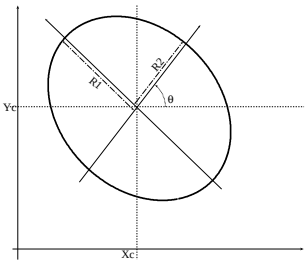

FIG. 4 – *Représentation des cinq paramètres caractérisant l'ellipse.*

sentation après échantillonnage sera sensiblement la même, puisque les valeurs enregistrées sont positionnées dans un repère lié à l'ellipse et donc au motif.
De plus, pour garantir une certaine insensibilité aux changements des conditions d'éclairement de la scène, le vecteur, une fois échantillonné, est classiquement centré et normé; ce qui permet de compenser des variations affines de la luminance entre l'image de référence et l'image courante.
Le choix de prendre une ellipse comme base d'échantillonnage est arbitraire. N'importe quelle autre forme peut convenir, à partir du moment où une procédure d'échantillonnage, garantissant les invariances aux changements de position, orientation et échelle, est définie.
Pour notre part, la position et la forme de l'ellipse sont, bien entendu, décrites par un vecteur à cinq paramètres correspondant à la position du centre (Xc, Yc), l'orientation (θ) et la longueur du grand et du petit axe $(R1, R2)$. L'interprétation géométrique de ces cinq paramètres est rappelée Figure 4.

3 Matrice d'interaction

3.1 Principes

Nous venons de voir que le motif que l'on désire suivre est inscrit dans une ellipse dont la position dans l'image est donnée par le vecteur de paramètres $\mathbf{E}$ avec $\mathbf{E} = (Xc, Yc, R1, R2, \theta)$.
Plus précisément, notons $\mathbf{E_p}$ le vecteur de paramètres prédit, $\mathbf{E_r}$ la position réelle du motif et

$$\mathbf{\Delta E} = \mathbf{E_r} - \mathbf{E_p}$$

la différence entre ces deux vecteurs (les vecteurs et matrices sont représentés en caractères gras).
D'un autre côté, le motif visuel à l'intérieur de l'ellipse prédite est échantillonné pour obtenir le vecteur de forme courant $\mathbf{I_c}$. Le vecteur de forme représentatif du motif à suivre (motif de référence) est quant à lui noté $\mathbf{I_{ref}}$. Posons :

$$\mathbf{\Delta I} = \mathbf{I_{ref}} - \mathbf{I_c}$$

Il est alors intéressant de savoir si l'on peut déterminer $\mathbf{\Delta E}$ en connaissant $\mathbf{\Delta I}$. Si c'est le cas, cela signifie qu'en mesurant la différence $\mathbf{\Delta I}$ entre le motif souhaité et le motif contenu dans l'ellipse prédite $\mathbf{E_p}$, on peut déterminer la correction $\mathbf{\Delta E}$ à apporter à la prédiction pour obtenir la position réelle du motif :

$$\mathbf{E_r} = \mathbf{E_p} + \mathbf{\Delta E}$$

3.2 Notion de matrice d'interaction

Nous pouvons raisonner comme si la variation de chacun des paramètres (ou erreur de prédiction) de l'ellipse était une fonction de la variation du motif visuel échantillonné (différence de motif observée).

$$\mathbf{\Delta E} = (\Delta Xc, \Delta Yc, \Delta R1, \Delta R2, \Delta\theta)$$

avec :

$$\begin{aligned}
\Delta Xc &= f_{Xc}(\mathbf{\Delta I}) \\
\Delta Yc &= f_{Yc}(\mathbf{\Delta I}) \\
\Delta R1 &= f_{R1}(\mathbf{\Delta I}) \\
\Delta R2 &= f_{R2}(\mathbf{\Delta I}) \\
\Delta\theta &= f_{\theta}(\mathbf{\Delta I})
\end{aligned}$$

L'idée consiste alors a approximer chacune des fonctions par leur développement au premier ordre. Cela revient à écrire :

$$\begin{aligned}
\Delta Xc &= [\frac{\partial f_{Xc}}{\partial \mathbf{I}}] \times \mathbf{\Delta I} \\
\Delta Yc &= [\frac{\partial f_{Yc}}{\partial \mathbf{I}}] \times \mathbf{\Delta I} \\
\Delta R1 &= [\frac{\partial f_{R1}}{\partial \mathbf{I}}] \times \mathbf{\Delta I} \\
\Delta R2 &= [\frac{\partial f_{R2}}{\partial \mathbf{I}}] \times \mathbf{\Delta I} \\
\Delta\theta &= [\frac{\partial f_{\theta}}{\partial \mathbf{I}}] \times \mathbf{\Delta I}
\end{aligned}$$

Avec la notation :

$$[\frac{\partial f_{Xc}}{\partial \mathbf{I}}] = (\frac{\partial f_{Xc}}{\partial i_1}, \ldots, \frac{\partial f_{Xc}}{\partial i_n})^t$$

où $\mathbf{I} = (i_1, \ldots, i_n)$ représente le vecteur échantillonné et où ^{t} dénote la transposition.
L'ensemble de ces 5 équations peut se noter sous la forme matricielle :

$$\mathbf{\Delta E^t} = \mathbf{A} \times \mathbf{\Delta I}$$

avec

$$\mathbf{A} = \left([\frac{\partial f_{Xc}}{\partial \mathbf{I}}], [\frac{\partial f_{Yc}}{\partial \mathbf{I}}], [\frac{\partial f_{R1}}{\partial \mathbf{I}}], [\frac{\partial f_{R2}}{\partial \mathbf{I}}], [\frac{\partial f_{\theta}}{\partial \mathbf{I}}] \right)^t$$

ou encore sous la notation :

$$\mathbf{A} = (\mathbf{AXc}, \mathbf{AYc}, \mathbf{AR1}, \mathbf{AR2}, \mathbf{A}\theta)^t$$

Dans le cas général, si N est la taille des vecteurs décrivant l'image et P le nombre de paramètres utilisés pour décrire la zone suivie dans l'image (ici $P = 5$), La matrice $\mathbf{A}$ est de taille $P \times N$.
Nous appelons par la suite cette matrice $\mathbf{A}$ *matrice d'interaction.*

3.3 Estimation de la matrice d'interaction

Nous proposons d'estimer la matrice d'interaction au moyen d'une phase d'apprentissage. Au début de cette phase, une ellipse est placée manuellement par l'utilisateur sur le motif à suivre. L'ellipse est ensuite aléatoirement déformée (modification de ses cinq paramètres) au voisinage du vecteur de référence. Pour chaque déformation, les variations du vecteur de paramètres de l'ellipse ainsi que les variations du motif échantillonné à l'intérieur de l'ellipse sont mémorisées. Ainsi, si l'on prend M mesures de ce type, N étant la taille du vecteur représentant le motif, il est possible d'estimer $\mathbf{A}$ si $M > N$. Cela revient à résoudre P systèmes d'équations à N inconnues. Nous estimons $\mathbf{A}$ par une minimisation au sens des moindres carrés, en utilisant un algorithme basé sur une décomposition en valeurs singulières.
En réalité, la résolution d'un seul système linéaire, ou plus exactement le calcul d'une seule matrice pseudo inverse est nécessaire. En effet, notons $\mathbf{\Delta I^j} = (\Delta i_1^j, \Delta i_2^j, ..., \Delta i_N^j)$ le vecteur de différence entre le motif de référence et le motif correspondant à la j^{ieme} perturbation et $\Delta\theta^j$ la variation d'orientation entre les ellipses permettant de calculer ces deux vecteurs.
Pour obtenir la ligne $\mathbf{A}\theta$ de la matrice d'interaction relative à l'orientation de l'ellipse, nous arrivons au système linéaire suivant :

$$\begin{bmatrix} \Delta i_1^1 & \Delta i_2^1 & . & \Delta i_N^1 \\ \Delta i_1^2 & \Delta i_2^2 & . & \Delta i_N^2 \\ . & . & . & . \\ \Delta i_1^j & \Delta i_2^j & . & \Delta i_N^j \\ . & . & . & . \\ \Delta i_1^M & \Delta i_2^M & . & \Delta i_N^M \end{bmatrix} \times \begin{bmatrix} \mathbf{A\theta_1} \\ \mathbf{A\theta_2} \\ \vdots \\ \mathbf{A\theta_N} \end{bmatrix} = \begin{bmatrix} \Delta\theta^1 \\ \Delta\theta^2 \\ . \\ \Delta\theta^j \\ . \\ \Delta\theta^M \end{bmatrix}$$

Pouvant se noter sous la forme matricielle suivante :

$$\mathbf{M_{\Delta I}} \times \mathbf{A}\theta = \mathbf{\Delta}\theta$$

La solution est alors obtenue par :

$$\mathbf{A}\theta = (\mathbf{M^t_{\Delta I}} \times \mathbf{M_{\Delta I}})^{-1} \times \mathbf{M^t_{\Delta I}} \times \Delta\theta = \mathbf{M^+_{\Delta I}} \times \Delta\theta$$

La matrice $\mathbf{M^+_{\Delta I}}$ est qualifiée de pseudo inverse de la matrice $\mathbf{M_{\Delta I}}$. Il est évident que le calcul des quatre autres lignes de la matrice d'interaction A utilise le produit de la même matrice avec des vecteurs de perturbation différents $(\Delta Xc, \Delta Yc, \Delta R1, \Delta R2)$.

3.4 Utilisation de la matrice d'interaction

Comme indiqué précédemment l'utilisation de la matrice d'interaction est relativement simple. Son produit par le vecteur correspondant à la différence entre le motif de référence et le motif courant donne un vecteur de correction à appliquer au paramètre de l'ellipse courante.

$$\mathbf{\Delta E^t} = \mathbf{A} \times \mathbf{\Delta} I$$

Il faut toutefois noter que les paramètres de correction (ou de déformation durant la phase d'apprentissage) doivent être choisi sous une représentation intrinséquement liée à l'ellipse pour que ces derniers soient indépendant de la position de cette dernière. Si, de ce point de vue, l'orientation et la longueur des axes ne posent aucun problème, il n'en est pas de même pour les translations du centre de l'ellipse. Pour cette raison, nous avons choisi de paramétrer les variations de ce dernier dans le repère défini par les deux axes principaux de l'ellipse courante.

3.5 Résultats expérimentaux

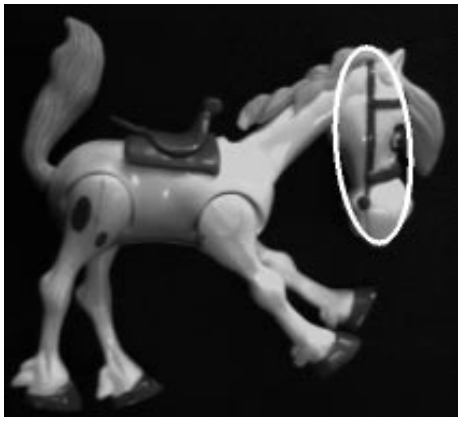

FIG. 5 – *Image utilisée pour l'étude de la convergence sur image statique. L'ellipse indique le motif de référence.*

Avant d'utiliser cette matrice d'interaction dans une application de suivi sur une séquence d'image, nous allons chercher à caractériser ses performances sur des images statiques. Dans ce cas, l'objectif est d'observer comment et dans quelles limites la matrice d'interaction permet de revenir sur le motif sélectionné lorsque l'on écarte l'ellipse de la position de référence.
L'image utilisée pour cet exemple est présentée Figure 5. L'ellipse tracée sur cette dernière englobe le motif visuel de référence choisi. Ce dernier est échantillonné sur 373 points à l'intérieur de l'ellipse selon la procédure décrite préalablement.
Une série de 1000 déformations aléatoires de l'ellipse de référence est réalisée pour estimer la matrice d'in-

teraction. Les amplitudes des variations des paramètres de l'ellipse sont de 15% de la longueur des axes pour la position du centre (rappelons que les variations de ce dernier sont réalisées le long des axes principaux de l'ellipse), de 10% sur la longueur des axes, et de 10° sur la rotation.
Nous présentons dans les sous sections suivantes des courbes donnant la valeur de la correction estimée (par le calcul de $\mathbf{A} \times \mathbf{\Delta I}$) en fonction de la variation $\mathbf{\Delta E}$ réellement effectuée.
Afin de simplifier la représentation, nous avons différencié deux cas : un cas où les erreurs de translations sont prépondérantes sur les erreurs liée à la longueur des axes et l'orientation de l'ellipse, et un cas inverse.

3.6 Translations

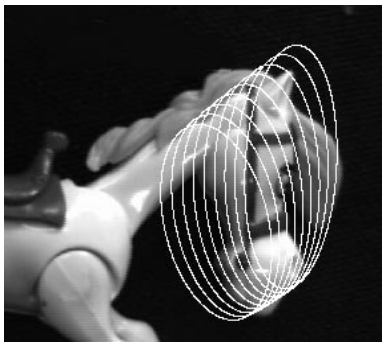

FIG. 6 – *Tests de translations : différentes ellipses sont calculées en translatant essentiellement le centre de l'ellipse de référence.*

La Figure 6 présente différentes ellipses, déduites de la position de référence principalement par une translation du centre (Xc, Yc). Afin de ne pas surcharger la figure, toutes les positions intermédiaires ne sont pas représentées. Toutefois, pour chacune d'entre elles, la matrice d'interaction est utilisée pour calculer la correction à apporter aux paramètres de l'ellipse déformée

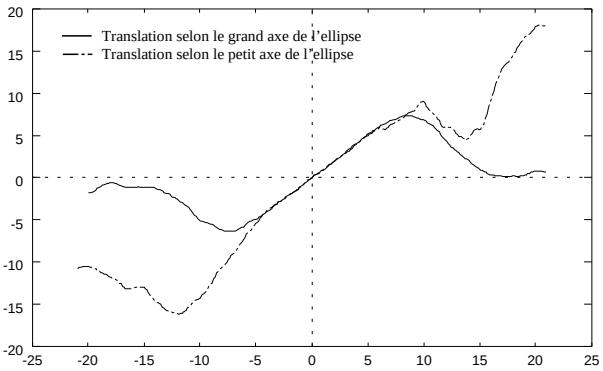

FIG. 7 – *Translation estimée en fonction de la translation réelle de l'ellipse (en pixels).*

afin de retomber sur l'ellipse de référence.
Ainsi, les courbes présentées Figure 7 indiquent la translation estimée en fonction de la perturbation réelle de l'ellipse.

3.7 Rotations et changement d'échelle

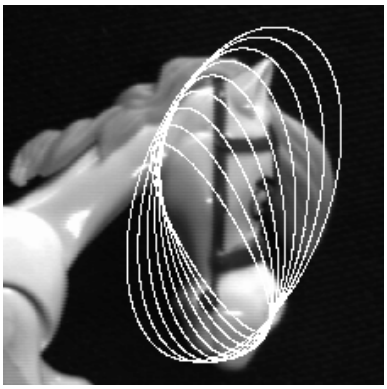

FIG. 8 – *Tests de rotation/changement d'échelle : différentes ellipses déformées majoritairement en tournant et grossissant l'ellipse de référence.*

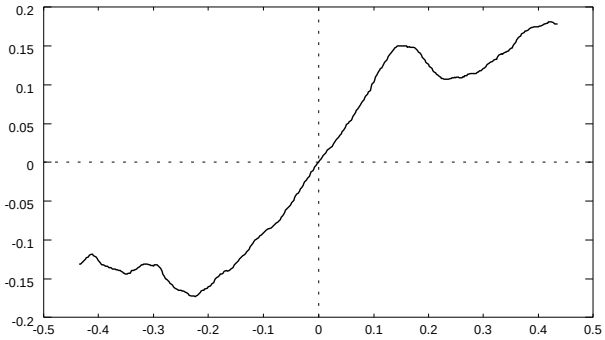

FIG. 9 – *Rotation estimée en fonction de la perturbation en rotation de l'ellipse (en radians).*

La Figure 8 présente différentes ellipses, écartées de la position de référence principalement par une combinaison de rotations et changements d'échelles (les positions intermédiaires ne sont pas toutes représentées, mais elles ont été prises en compte pour les calculs).
Les courbes présentées Figure 9 et Figure 10 indiquent les rotations et changements d'échelle estimés en fonction des variations réelles.

3.8 Analyse des courbes - Domaine de convergence

Autour des petites variations, la variation prédite correspond bien à la variation effectuée. C'est la partie linéaire centrale de la courbe. Dans ce domaine l'approximation du premier ordre est satisfaisante. Il suffira d'une itération pour passer de la position prédite du motif à sa position réelle dans l'image. Ce domaine

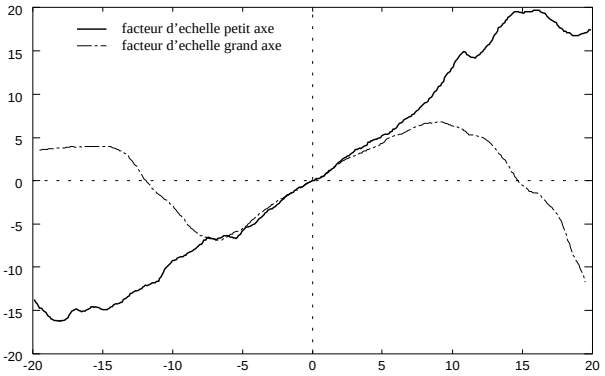

FIG. 10 – *Longueur des axes estimée en fonction de la perturbation de la longueur des axes (en % de la longueur).*

correspond à des variations d'environ +/-10 pixels en translation, +/-10% de la longueur des axes et de +/-10 degrés en rotation (Notons que ceci correspond approximativement à l'ordre de grandeur de l'écart type utilisé pour perturber l'ellipse de référence lors de l'apprentissage de la matrice d'interaction).

Il nous faut toutefois mentionner que le domaine de convergence est plus vaste que cette partie linéaire de la courbe. En effet, il suffit que le signe soit correct pour que la correction s'opère dans le bon sens. Il faudra dans un tel cas, itérer quelques fois le processus pour se ramener dans la zone linéaire, où une dernière itération placera l'ellipse sur le motif de référence. Le domaine de convergence observé est environ deux fois plus grand que le domaine de linéarité.

Bien entendu, à chaque itération, il est nécessaire de calculer le $\mathbf{\Delta I}$ correspondant à la nouvelle position de l'ellipse.

Ces tests ont été effectués sur différents types d'images, avec différents types de variations. Les exemples que nous venons de présenter sont représentatifs des résultats obtenus pour l'ensemble des expériences réalisées.

4 Suivi de motifs texturés

4.1 Principes

Dans cette section, la technique préconisée est exploitée pour réaliser le suivi d'un motif texturé dans une séquence vidéo.

A partir d'une image du motif fournie par l'utilisateur, le système calcule hors ligne la matrice d'interaction. La méthode consiste tout d'abord à sélectionner manuellement et approximativement le motif à suivre dans la première image de la séquence. Il nous faut ensuite calculer la différence entre ce motif et le motif de référence, puis effectuer son produit par la matrice d'interaction pour obtenir un vecteur correctif sur la

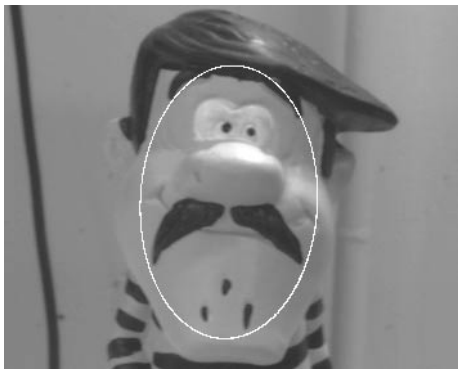

FIG. 11 – *Le motif à suivre est inscrit dans l'ellipse.*

position du motif; ces deux dernières étapes étant itérées sur chaque nouvelle image de la séquence.

Cet algorithme peut être décrit au moyen du pseudo-code suivant :

```
DEBUT Procédure SUIVI(E,Image)
   Ep = Prediction(E)
   I = Echantillonner(Image,EP)
   DI = I-Iref
   E = EP + A*DI
FIN Procédure SUIVI
```

Du point de vue coût algorithmique, la méthode consiste donc principalement à soustraite deux vecteurs et à faire un produit de matrices. Cela représente, pour un vecteur à 373 éléments, un temps de calcul total de 9ms sur une station de travail Silicon Graphics O_2 (station de base Silicon Graphics).

Compte tenu de son efficacité, l'implémentation de l'algorithme proposé permet de suivre en temps réel un motif quelconque sur une station de travail classique. La fréquence des traitements étant importante par rapport à la vitesse des objets dans les séquences traitées, nous n'avons pas eu besoin d'utiliser d'algorithme de prédiction de mouvement. En effet, dans les séquences traitées, l'écart de position du motif entre deux images successives reste compatible avec la zone de convergence de l'algorithme. La position prédite est donc celle qu'occupait le motif dans l'image précédente. Pour des déplacements plus rapides, l'utilisation d'un algorithme d'estimation et de prédiction du mouvement serait bien entendu nécessaire.

4.2 Résultats

Dans l'expérience présentée, le motif à suivre est celui donné Figure 11. Etant donné la nature des données traitées (un flot d'image vidéo), les résultats sont difficiles à représenter. Nous avons choisi de visualiser les résultats de suivi correspondant à six instants différents de l'expérience (voir Figure 12) [1]

1. Vidéo disponible sur le web : http://wwwlasmea.univ-bpclermont.fr/Personnel/Frederic.Jurie/rfia2000.html

FIG. 12 – *Illustration du suivi de motif : le motif suivi est inscrit dans l'ellipse blanche.*

Les images retenues ont été sélectionnées de manière à montrer la robustesse de l'algorithme face à des rotations planaires, à de forts changements d'échelle, et également aux déformations du motif du à la nature volumique de l'objet suivi. En effet, dans les trois dernières images ont peut constater que l'algorithme arrive très bien à suivre le motif, même si celui-ci est fortement déformé par une orientation de l'objet par rapport à la caméra différente de celle caractérisant l'image de référence (voir Figure 11). Nous avons observé que l'algorithme pouvait tolérer des inclinaisons allant jusqu'à 45 degrés.

5 Suivi d'objets 3D

5.1 Principes

Nous proposons d'utiliser l'algorithme de suivi proposé dans une application de suivi d'objets 3D. La principale difficulté par rapport à la technique présentée précédemment revient à gérer la variation du motif à suivre en fonction de l'attitude relative objet/caméra. Deux possibilités se présentent pour traiter ce problème :

- estimer en permanence l'attitude 3D de l'objet de façon à pouvoir prévoir à chaque instant l'aspect du motif à suivre.

- disposer d'une collection d'aspects potentiels de l'objet, et sélectionner en permanence l'aspect compatible avec l'image courante (et donc indirectement avec l'attitude de l'objet).

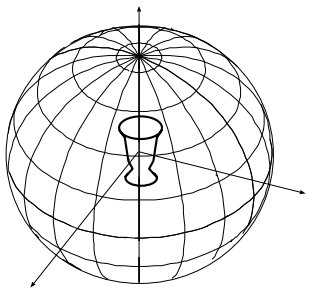

FIG. 13 – *Construction du modèle : les vues de l'objet sont obtenues en déplaçant la caméra sur une sphère centrée sur l'objet.*

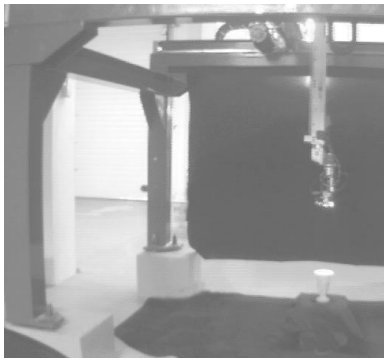

FIG. 14 – *La sphère de vues est obtenue au moyen d'une caméra embarquée sur un bras robotique.*

Nous préconisons la seconde famille d'approche. En effet, bien que plus coûteuse en terme de sauvegarde (il est nécessaire de stocker plusieurs centaines de vues de chaque objet à suivre), elle évite le recours à un modèle géométrique tridimensionnel et texturé de l'objet. De plus cela dispense d'avoir à calculer, à chaque image, une localisation 3D de l'objet ; une telle estimation n'étant pas toujours facile dans le cas d'objets volumiques texturés de forme complexe.

5.2 Construction du modèle de l'objet

Le modèle de l'objet est donc constitué d'une collection d'aspects. Les vues correspondantes sont obtenues en déplaçant la caméra sur une sphère centrée sur l'objet (voir Figure 13); les points d'acquisition étant approximativement équi-répartis. Nous appelons l'ensemble de ces images : *sphère de vues.*

En pratique, cette série d'acquisitions se fait à l'aide d'un bras robotique équipé d'une caméra embarquée en bout d'effecteur (voir Figure 14).

5.3 Suivi de la vue dans l'image & Suivi des vues sur la sphère de vues

Deux types de suivi différents doivent être menés simultanément :

- un suivi de la vue actuelle de l'objet dans l'image, permettant de compenser les trois translations de l'objet (la translation selon l'axe optique revenant à un changement d'échelle) ainsi que la rotation de l'objet autour de l'axe optique de la caméra (rotation planaire du motif). Ce suivi correspond à l'algorithme de suivi proposé dans la section précédente.
- un suivi de la vue courante sur la sphère de vues, pour compenser les deux autres degrés de liberté rotoïdes de l'objet.

Le premier point ayant déjà été développé, il nous reste à décrire le second, c'est à dire celui du suivi de l'aspect de l'objet sur la sphère de vues.
Soit V_0 l'aspect de l'objet observé dans l'image précédente, et soit $CV = \{V_0, V_1, \ldots, V_n\}$ l'ensemble de vues constitué de V_0 et de ses n vues voisines. La notion de voisinage est établie à partir d'une notion de distance sur la sphère de vues. Suivre l'aspect de l'objet sur la sphère de vues revient donc à déterminer quelle est la vue de CV la plus proche de la vue actuellement observée dans l'image.
Pour atteindre cet objectif, nous utilisons les matrices d'interaction calculées pour chacune des vues. Soit $A(V_i)$ la matrice d'interaction de la vue V_i et $I(V_i)$ le vecteur de forme correspondant à la vue V_i. La recherche de la vue la plus proche se fait au moyen de l'algorithme suivant (où `NORME` retourne la norme d'un vecteur) :

```
DEBUT Procédure SUIVI_VUE(E,Image,V)
   CV = CalcVoisinage(V,n)
   I = Echantillonner(Image,E)
      POUR (i=0; i<n; i++)
        Etest = E + A(Vi) * (I-I(Vi))
        Itest = Echantillonner(Image,Eest)
        Diff = Norme (I(Vi)-Itest)
        SI (Diff<MIN) ALORS
            Diff=MIN
            NouvelleVue=Vi
        FIN SI
      FIN POUR
   V = NouvelleVue
FIN Procédure SUIVI
```

Cela signifie que pour chaque vue du voisinage, nous recalons le motif de l'image courante sur son motif de référence en utilisant sa propre matrice d'interaction, puis une fois le motif recalé nous le comparons au motif de référence. La vue conduisant à l'erreur la plus faible devient la nouvelle vue courante.

FIG. 15 – *Objet utilisé pour les expérimentations de suivi d'objets 3D.*

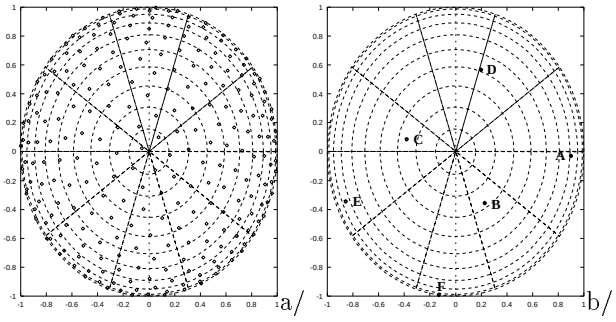

FIG. 16 – *Représentation de la sphère de vues et des 250 points d'acquisition d'images pour modéliser la sphère de vue (vue de dessus).*

5.4 Expérimentations

Nous présentons une série d'expériences portant sur le suivi de l'objet présenté Figure 15. C'est un objet dont le suivi 3D présente des difficultés car les rotations selon son axe de révolution entrainent un faible changement de son aspect.
Ce vase est modélisé à partir de 250 vues prises sur la demi-sphère supérieure (l'objet n'est jamais vu par en dessous).
Les positions des 250 vues acquises sur la sphère sont représentées par un point sur la Figure 16-a/ (la sphère étant vue de dessus). La figure 16-b/ met en avant, à titre d'exemple, 6 vues particulières. Ces six aspects de l'objet (vecteurs de forme de référence) sont représentés Figure 17.

5.5 Résultats

Nous ne présentons que des résultats concernant le suivi sur la sphère de vue, des résultats concernant le suivi dans l'image ayant déjà été présentés (voir section 4).
Pour ce faire, nous avons fait effectuer à la caméra une trajectoire sur la même demi sphère centrée sur l'objet et noté précisément la position de la caméra à chaque instant de prise de vue.
La figure 18 présente la trajectoire réellement effectuée sur la demi sphère supérieure(traits pleins) ainsi que la trajectoire estimée par l'algorithme à partir des

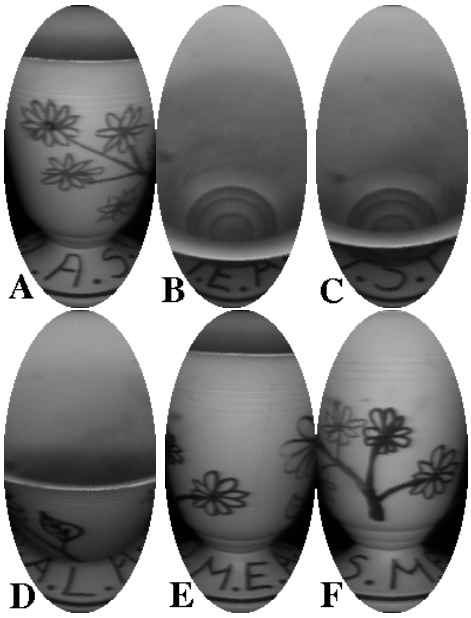

FIG. 17 – *Les six vues représentées sur la Figure 16-b/*

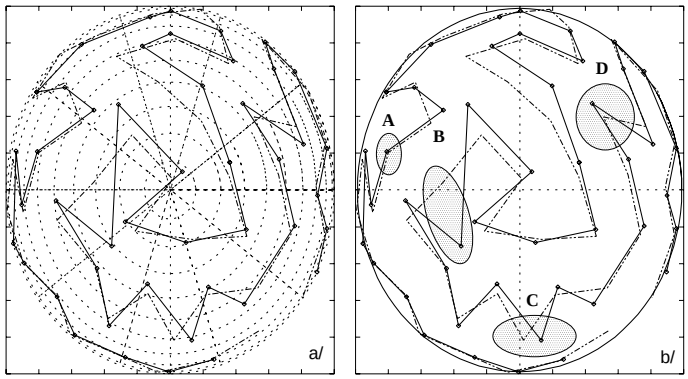

FIG. 18 – *a/Trajectoire effectuée sur la demi sphère supérieure (traits pleins). Trajectoire estimée (traits discontinus).b/Mise en exergue de quatre positions particulières le long de la trajectoire.*

aspects stockés dans la sphère de vue (traits discontinus). L'estimation correspond assez bien à la réalité, sauf en deux ou trois points où l'erreur semble importante.

Afin de mieux analyser ces erreurs, nous allons nous intéresser à quatre positions particulières, notées A, B, C et D, représentées la figure 18-b/. La figure 19 donne, pour chacune de ces quatre positions particulières, l'aspect de l'objet dans l'image ainsi que la vue considérée comme la plus proche. Ainsi on s'aperçoit que même si pour la position B la position estimée sur la sphère de vue semble relativement éloignée de la réalité, l'aspect de la vue sélectionnée est très proche de la réalité. Cela est dû au fait que pour les vues voisines du sommet de la sphère, une rotation de l'objet autour de son axe de révolution n'entraîne que de très faibles modifications de son aspect, de par l'uniformité de sa surface interne.

La figure 20 représente quatre images issues d'une expérience pendant laquelle l'objet est déplacé manuellement devant la caméra. Le suivi s'avère là encore très robuste, ce qui montre la tolérance, par rapport à l'expérience précédente, face au changement d'échelle entre le motif courant et ceux stockés dans la sphère de vue.

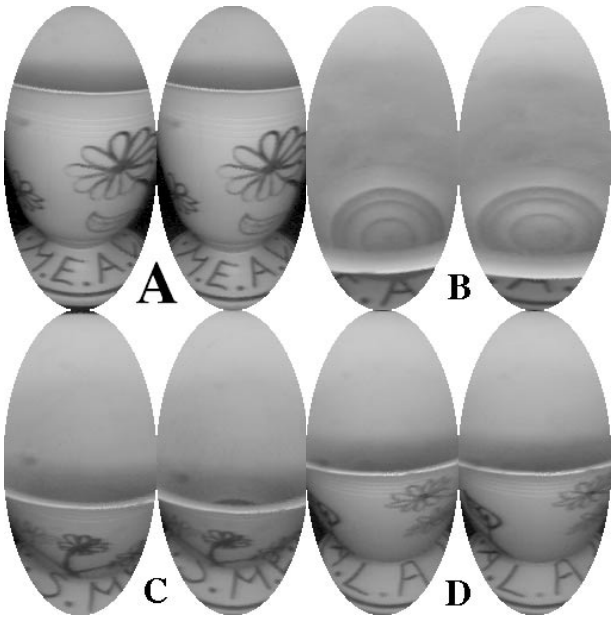

FIG. 19 – *Représentation de l'aspect de l'objet dans l'image (à gauche) et de la vue la plus proche sur la sphère de vues (à droite), pour les position A,B,C et D de la trajectoire. Ces quatres positions sont celles représentées sur la Figure 18.*

6 Conclusions

Nous avons présenté dans cet article une technique efficace de suivi d'objets planaires ou volumiques dans des séquences d'images. L'originalité de cette dernière réside dans l'idée d'estimer une correction à apporter sur la position prédite du motif recherché. Cette correction est le résultat du simple produit d'une matrice,

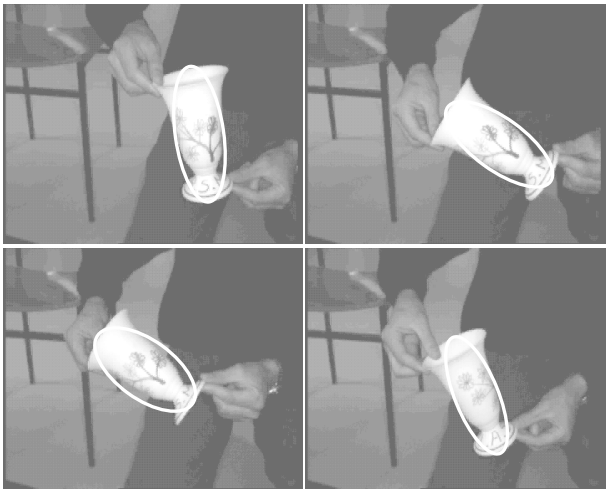

FIG. 20 – *Illustration du suvi d'objet. L'ellipse représente la position supposée de l'objet dans l'image à différents instants de la séquence.*

appelée matrice d'interaction, avec la différence entre le motif observé et celui de référence.
Cette technique nous a permis de mettre au point un algorithme de suivi de motif 2D temps réel. Nous l'avons également appliquée au suivi d'objet 3D, en la combinant avec un algorithme permettant de gérer le changement du motif à suivre. Des résultats expérimentaux nous ont permis de valider les différentes approches préconisées et d'apprécier le grande robustesse induite par l'utilisation des matrices d'interactions.
Nous travaillons actuellement sur des techniques permettant de rendre cette approche résistante aux occultations partielles du motif recherché.

Références

[1] M.J. Black and A.D. Jepson. Eigentracking : Robust matching and tracking of articulated objects using a view-based representation. In *Proc. European Conference on Computer Vision*, volume I, pages 329–342, Cambridge, UK, 1996.

[2] R. Brunelli and Poggio T. Template matching : Matched spatial filter and beyond. A.I. Memo 1549, M.I.T., October 1995.

[3] J.L Crowley, P. Stelmaszyl, and C. Discours. Measuring image flow by tracking edge-lines. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 658–664, San Diego, California, 1989.

[4] T. Darrell, I.A. Essa, and Pentland A.P. Task-specific gesture analysis in real-time using interpolated views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(12):1236–1242, 1996.

[5] C. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 147–151, Mancherster, 1988.

[6] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. European Conference on Computer Vision*, volume I, pages 343–356, Cambridge, UK, 1996.

[7] F. Jurie. Tracking objects with a recognition algorithm. *Pattern Recognition Letters*, 3-4(19):331–340, 1998.

[8] H. Kollnig and H.H. Nagel. 3d pose estimation by directly matching polyhedral models to gray value gradients. *International Journal of Computer Vision*, 23(3):283–302, 1997.

[9] D.G. Lowe. Robust model-based motion tracking through the integration of search and estimation. *International Journal of Computer Vision*, 8(2):113–122, 1992.

[10] B. Moghaddam and A. Pentland. A subspace method for maximum likelihood taget detection. Technical report, M.I.T. Media Laboratory Perceptual Computing Section, 1995.

[11] F. Pla and J. Marchant. Matching feature points in image sequences through a region based method. *Computer Vision and Image Understanding*, 66(3):271–285, 1997.

[12] J.J Wu, R.E. Rink, T.M. Caelli, and V.G. Gourishankar. Recovery of the 3-d location and motion of a rigid object through camera image. *International Journal of Computer Vision*, 3:373–394, 1989.

[13] G.S. Young and R. Chellappa. 3d motion estimation using a sequence of noisy stereo images. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 710–716, Ann Arbor, Michigan, 1988.