



Suivi temps-réel d'objet plan: approche hybride contour/texture

Real-time approach for model-free hybrid 2D tracking

M.Pressigout¹

E.Marchand¹

¹ Lagadic, IRISA/INRIA Rennes

IRISA, Campus de Beaulieu, Rennes

Muriel.Pressigout@irisa.fr ou Éric.Marchand@irisa.fr

Résumé

Le suivi temps-réel d'un objet dans une séquence d'images reste un problème sensible quand il s'agit d'obtenir des résultats précis et de prendre en compte d'occultations. La méthode décrite dans cet article permet un suivi plus efficace d'objets planaires sans utiliser de marqueurs spécifiques. Elle se base sur la fusion d'informations visuelles et sur l'estimation d'une transformation 2D. Les paramètres de cette transformation sont estimés par une minimisation itérative d'un critère hybride qui intègre à la fois des informations sur la texture et sur le contour de l'objet suivi. L'algorithme est alors plus robuste et permet d'achever un suivi correct quand l'utilisation d'un seul type d'information n'aurait pas suffi à obtenir un résultat satisfaisant. Ce suivi hybride a été développé pour des objets dont le contour peut être modélisé par des lignes mais aussi par une B-Spline. Dans ce dernier cas, l'implémentation est réalisée en utilisant les NURBS pour diminuer fortement le temps de calcul. L'efficacité de ce suivi a été testée sur des séquences d'images ainsi que lors d'expériences d'asservissement visuel avec une caméra embarquée sur un robot.

Mots Clef

Fusion d'informations, suivi contour/texture, suivi robuste, NURBS.

Abstract

The robustness and accuracy are major issues in real-time tracking. This paper describes a reliable tracking for markerless objects based on the fusion of multimodal visual cues and on the estimation of a 2D transformation. The parameters of this transformation are estimated using non-linear minimization of a unique criterion that integrates information on both texture and contour of the tracked object. The proposed tracker is then more robust and succeeds in conditions where methods based on a single cue fail. The tracker can deal with polygonal shaped objects but also with those which can be modeled by a B-spline. In the lat-

ter case, NURBS are used to reduce time processing. The efficiency and the robustness of the proposed method are tested on image sequences as well as during image-based servoing control experiments.

Keywords

Multimodal tracking, edge and correlation-based tracking, robust tracking, NURBS.

1 Introduction

L'élaboration d'algorithmes de suivi en temps-réel d'objets dans des séquences d'images est une problématique majeure pour de nombreuses applications liées à la vision par ordinateur, la robotique, l'asservissement visuel, *etc.* Un processus fiable d'extraction puis de suivi spatio-temporel de l'information visuelle est en effet une des clés du succès, ou de l'échec, de telles applications. Il apparaît d'autre part primordial pour introduire les techniques de suivi dans un large éventail d'applications, de pouvoir appréhender des scènes naturelles, c'est-à-dire, sans marqueurs, avec des objets polyédriques ou non, et des conditions d'illumination variables, . . . Différentes techniques existent pour parvenir à cet objectif. Schématiquement elles peuvent être divisées en deux grandes familles: celles basées sur le contour et celles basées sur la texture de l'objet. La première approche consiste essentiellement à suivre des primitives dans l'espace image ou 3D comme des primitives géométriques (points, lignes, cercles, . . .), le contour de l'objet, la projection des contours d'un objet 3D, *etc.* La dernière utilise un critère de corrélation lié à l'information donnée par les niveaux de gris du motif de l'objet ou d'autres informations présentes dans ce motif (couleur, . . .).

Le suivi basé contour repose sur les forts gradient spatiaux délimitant le contour de l'objet ou certaines primitives géométriques présentes dans un motif (points, lignes, distances, splines, . . .). En ce qui concerne le suivi dans l'espace de l'image (suivi 2D), cette approche consiste à décrire l'objet à suivre à l'aide de primitives

géométriques comme des points particuliers [21, 29], des angles, des contours [3, 4], des segments de droites [5, 15] ou des ellipses [31], etc. Les contours actifs ou snakes sont également basés sur de tels gradients et peuvent être également utilisés pour déterminer une forme complexe [4]. La connaissance d'un modèle 3D de l'objet à suivre [10, 11] transforme le problème de suivi de contour en un problème de calcul de pose. Les principaux avantages d'une telle approche sont la robustesse et la performance améliorées par l'introduction d'informations 3D sur la scène. La gestion de données erronées dans le processus de suivi. Cependant, ces travaux ne seront pas détaillés puisque les primitives géométriques considérées dans cet article sont uniquement celles extraites dans l'image, la connaissance du modèle 3D de l'objet étant une contrainte que nous avons voulu éviter dans ces travaux. Les techniques de suivi basé contour sont très efficaces pour des applications nécessitant un suivi rapide mais sont cependant très sensibles aux environnements texturés qui sont une cause d'échec fréquente.

Les approches précédentes reposent essentiellement sur une analyse des gradients d'intensité dans les images. Quand la scène est trop complexe (par exemple quand l'environnement est trop texturé ou si les contours de l'objet sont difficilement détectables), il est nécessaire d'utiliser d'autres méthodes. Une solution est alors de considérer directement le signal lumineux dans les images et d'effectuer directement la mise en correspondance 2D sur une partie de l'image sans passer par une phase d'extraction de primitives. Les problèmes de suivi reposant sur l'apparence ("*template-based matching*") ou d'estimation du mouvement apparent rentrent dans cette catégorie. L'objectif de tels algorithmes est d'estimer un ensemble de paramètres qui décrivent au mieux la transformation ou le mouvement de la partie de l'image considérée en optimisant un certain critère de corrélation. La recherche exhaustive de la transformation qui minimise ce critère n'est pas une solution efficace. Il existe des techniques de minimisation capables de résoudre ce problème en prenant en compte des modèles de transformations assez complexes (comme des mouvements affines ou homographiques par exemple). Dans [14], les auteurs ont proposé une approche basée sur ce principe en considérant la variation des paramètres du modèle de mouvement comme une fonction linéaire des différences d'intensité dans l'image. Ils définissent une matrice Jacobienne qui lie les variations des paramètres de la transformation aux variations de l'intensité lumineuse. Des extensions de cette approche ont été proposées. Dans [19], la pseudo-inverse de cette matrice est apprise lors d'une phase hors-ligne et [2, 7, 19] estiment une homographie alors que dans [14] le modèle du mouvement est affine. En outre, [2] reprend cette formulation en utilisant une minimisation du second ordre basée sur l'algèbre de Lie pour accélérer le processus de minimisation en annulant les termes du second ordre. Le "*template*" utilisé comme référence peut être mis à jour [24, 25] ou non [14] pour accélérer le suivi.

Il faut noter que ces méthodes sont étroitement liées aux algorithmes classiques d'estimation du mouvement [26]. De telles méthodes de suivi sont rapides et robustes quand la texture de l'objet se prête bien à une telle analyse mais manque parfois de précision.

Ces deux types d'approches présentent des avantages et des inconvénients complémentaires. Pour développer des algorithmes robustes aux données aberrantes, et donc à des potentielles occultations, il est intéressant de prendre en compte des informations visuelles liées à ces deux méthodes. Elles peuvent être exploitées séquentiellement de manière à combiner robustesse et précision comme dans [1, 9, 22]. Dans ce cas, l'estimation du mouvement est cependant principalement utilisée pour obtenir un meilleur recalage sur les contours (et assurer ainsi une plus grande robustesse du suivi). L'idée est ici d'utiliser en même temps les deux approches pour effectuer simultanément l'estimation de mouvement et le suivi de l'objet et ainsi mieux exploiter les avantages de chacun. D'autres approches se basent sur un cadre probabiliste. Dans [28], les auteurs utilisent une méthode basée texture pour trouver la projection dans l'image du contour d'un modèle 3D. Le calcul de la position la plus probable d'éléments texturés du contour remplace alors la détection standard basée sur les gradients. Des méthodes de suivi classiques exploitant un seul type de primitives visuelles, comme l'algorithme CONDENSATION, ont été étendues au suivi de primitives de natures diverses [18]. La fusion de primitives visuelles 2D a également été étudiée dans [20] et appliquée pour des applications d'asservissement visuel. Néanmoins, ce travail n'est pas directement lié à la fusion d'information basée contour ou texture comme c'est le cas dans cet article.

La méthode présentée ici intègre simultanément les deux approches. Puisque chacun des suivis peut être vu comme un problème d'optimisation, notre objectif est de définir un vecteur d'état unique qui permette de décrire aussi bien l'apparence du motif de l'objet que la position de ses contours. En considérant ce vecteur d'état, il est possible d'estimer les paramètres d'une transformation 2D qui minimise l'erreur entre le motif courant de l'objet regroupant différents types d'informations visuelles et celui qui contient les valeurs de référence associées. L'algorithme de suivi hybride fusionne l'estimation de mouvement de points de contours et de points de texture dans un unique processus de minimisation. Puisque les données sont éventuellement bruitées, il est nécessaire d'effectuer une minimisation robuste. Pour ce faire, l'introduction de M-estimateurs permet d'obtenir une implémentation de type IRLS (Iteratively Re-weighted Least Squares). Une approche similaire a été proposée par [23]. Dans ce dernier travail, l'algorithme de suivi basé texture est celui décrit dans [19] où le Jacobien est appris tandis que le nôtre est proche de celui présenté dans [14] où est utilisée une formulation analytique du Jacobien. Dans [23], les points de contours et de texture sont classifiés selon les valeurs propres de la matrice d'autocorrélation du

signal. Les contours forts sont alors marqués comme points de contour et, par conséquence, les points restants classés comme points de texture n'apportent que peu d'information puisqu'ils appartiennent à des régions de gradients assez lisses. L'optimisation d'un critère hybride a également été utilisée dans [30] dans le cadre du calcul de pose et de la réalité augmentée.

Les contributions apportées par les travaux présentés dans cet article sont la reformulation de suivis classiques, l'un basé sur les primitives géométriques définies par les contours de l'objet, l'autre basé sur son apparence, de façon à les réunifier dans un même cadre. Cette généralisation permet la fusion des deux suivis de manière assez directe et les résultats montrent l'intérêt d'une telle fusion. La méthode de suivi hybride que nous avons développée est présentée dans la section 2. Le cadre général du suivi basé sur une estimation d'une transformation 2D est développé dans la section 2.1 et deux méthodes de suivis, l'une basée contour et l'autre basée texture sont respectivement décrites dans les sections 2.2 et 2.3. La section 2.4 présente la fusion de ces deux suivis donnant un nouvel algorithme efficace. Pour finir, les résultats de la section 3 illustrent le comportement de cette nouvelle méthode sur des séquences d'images réelles mais aussi lors d'une expérience d'asservissement visuel.

2 Suivi basé sur l'estimation d'une transformation 2D

2.1 Estimation d'une transformation 2D

Cadre général. La transformation 2D qui relie la projection de l'objet entre l'image \mathbf{I}^{t-1} et l'image \mathbf{I}^t est telle que si $\mathbf{x}_{\mu_t} = (x_{\mu_t}, y_{\mu_t})^T$ est un point de \mathbf{I}^t appartenant à la projection de l'objet et $\mathbf{x}_{\mu_{t-1}}$ son correspondant dans \mathbf{I}^{t-1} , alors :

$$\mathbf{x}_{\mu_t} = \Psi_{\mu_t}(\mathbf{x}_{\mu_{t-1}}) \quad (1)$$

où Ψ_{μ_t} est la transformation 2D décrite par M paramètres. Classiquement, le modèle de la transformation utilisé est un modèle affine ou une homographie. Bien que les deux cas aient été étudiés, seule l'homographie est considérée ici puisque c'est un cas plus général. On a en coordonnées homogènes :

$$\mathbf{x}_t^h = \begin{pmatrix} \mu_0 & \mu_1 & \mu_2 \\ \mu_3 & \mu_4 & \mu_5 \\ \mu_6 & \mu_7 & \mu_8 \end{pmatrix} \mathbf{x}_{t-1}^h \quad (2)$$

Par conséquent, les paramètres à estimer sont :

$$\boldsymbol{\mu} = (\mu_0, \mu_1, \mu_2, \mu_3, \mu_4, \mu_5, \mu_6, \mu_7, \mu_8)^T \quad (3)$$

Il n'est pas requis de choisir une représentation spécifique de l'homographie qui est définie à un facteur près puisque la méthode proposée est invariante à ce facteur d'échelle.

Soit \mathbf{m}_{μ} le vecteur colonne de dimension N qui contient la valeur des primitives dans l'image estimée à partir des paramètres de la transformation 2D $\boldsymbol{\mu}$: $\mathbf{m}_{\mu} = (m_{\mu}^1, \dots, m_{\mu}^N)^T$. Dans la suite, m_{μ}^i sera soit les niveaux de gris $I^t(\mathbf{x}_{\mu}^i)$ observés aux points \mathbf{x}_{μ}^i dans \mathbf{I}^t [14], soit une distance entre un point de l'image et une primitive géométrique 2D, mais la méthode peut être élargie à d'autres primitives. Sa valeur de référence dans \mathbf{I}^t est notée $\mathbf{m}_{\mu_t^*}$ et sa valeur courante dans \mathbf{I}^t calculée à partir de la transformation estimée $\boldsymbol{\mu}_t$ est représentée par \mathbf{m}_{μ_t} . L'idée de base est d'estimer la transformation 2D qui vérifie (1). Ceci est réalisé en minimisant l'erreur entre la valeur courante \mathbf{m}_{μ_t} et la valeur $\mathbf{m}_{\mu_t^*}$ observée dans l'image courante \mathbf{I}^t :

$$\hat{\boldsymbol{\mu}}_t = \underset{\boldsymbol{\mu}_t}{\operatorname{argmin}} \|\mathbf{m}_{\mu_t} - \mathbf{m}_{\mu_t^*}\|^2 \quad (4)$$

Lorsque cette erreur est minimisée, les primitives estimées correspondent aux observations et la transformation 2D estimée à la transformation 2D réelle.

La continuité du mouvement garantit que $\boldsymbol{\mu}_t = \boldsymbol{\mu}_{t-1} + \delta\boldsymbol{\mu}$. Le problème est alors d'estimer les valeurs $\hat{\delta\boldsymbol{\mu}}$ qui minimisent l'erreur e définie par: $e = \|\mathbf{m}_{\boldsymbol{\mu}_{t-1} + \delta\boldsymbol{\mu}} - \mathbf{m}_{\mu_t^*}\|^2$. Une minimisation itérative basée sur une approximation du premier ordre minimise l'erreur e par :

$$\hat{\delta\boldsymbol{\mu}} = -\lambda \mathbf{J}_{\mathbf{m}_{\mu_t}}^+ e \quad (5)$$

λ est le facteur scalaire qui permet d'assurer une décroissance exponentielle de l'erreur, $\mathbf{J}_{\mathbf{m}_{\mu_t}}$ est la matrice Jacobienne de \mathbf{m} par rapport aux paramètres courants de la transformation 2D. C'est une matrice $N \times M$ contenant les N matrices Jacobiennes $\mathbf{J}_{\mathbf{m}_{\mu_t}^i}$ de chaque primitive visuelle $\mathbf{m}_{\mu_t}^i$:

$$\mathbf{J}_{\mathbf{m}_{\mu_t}} = (\mathbf{J}_{\mathbf{m}_{\mu_t}^1}, \dots, \mathbf{J}_{\mathbf{m}_{\mu_t}^N})^T \quad (6)$$

avec

$$\mathbf{J}_{\mathbf{m}_{\mu_t}^i} = \frac{\partial \mathbf{m}_{\mu_t}^i}{\partial \boldsymbol{\mu}_t} \quad (7)$$

Pour que le suivi se fasse le plus rapidement possible, la matrice Jacobienne peut être approximée par celle calculée à la première itération du processus de minimisation. Seule une convergence locale peut alors être obtenue. Cependant, comme le déplacement de la caméra entre deux images est supposé faible, c'est une solution efficace. Il sera vu dans la section des résultats que cette approximation reste valable pour des mouvements importants.

Estimation robuste. Dans une séquence vidéo, les données sont bruitées ou des occultations peuvent avoir lieu. Puisque le processus de minimisation est sensible à de telles erreurs, des M-estimateurs sont introduits dans (5) pour éliminer les données erronées:

$$\hat{\delta\boldsymbol{\mu}} = -\lambda (\mathbf{D}\mathbf{J}_{\mathbf{m}_{\mu_t}})^+ \mathbf{D}e \quad (8)$$

où \mathbf{D} est une matrice diagonale $N \times N$ telle que :

$$\mathbf{D} = \text{diag}(w_1, \dots, w_N) \quad (9)$$

Les N poids w_i reflètent la confiance que l'on a en chaque primitives visuelles $m_{\mu_t}^i$ [16] et sont en général donnés par :

$$w_i = \frac{\psi(\delta_i/\sigma)}{\delta_i/\sigma} \quad (10)$$

avec $\psi(u) = \frac{\partial \rho(u)}{\partial \mathbf{m}_{\mu_t}}$ (ψ est la fonction d'influence) et δ_i le résidu normalisé donné par $\delta_i = \Delta_i - \text{Med}(\Delta)$ (où $\text{Med}(\Delta)$ est la déviation standard des données correctes). De nombreuses fonctions d'influence sont utilisées dans la littérature. Nous avons retenu la fonction de Tukey car elle rejette complètement les données aberrantes.

L'approche décrite jusque-là est valable pour n'importe quelle primitive visuelle \mathbf{m} à partir du moment où la matrice Jacobienne associée $\mathbf{J}_{\mathbf{m}}$ est disponible. Ce cadre général est appliqué dans les deux paragraphes suivants, d'abord pour des primitives basées contour puis pour des primitives basées texture. Pour chacun des suivis, le choix des primitives et de la matrice Jacobienne est donné. La reformulation de ces deux suivis de base, assez classiques, peut être inhabituelle au lecteur mais elle est nécessaire pour faire ressortir le cadre général et permettre la fusion des deux algorithmes de manière assez directe.

2.2 Estimation d'une transformation 2D basée contour

Dans cet article, le suivi basé contour se base sur le suivi de points situés sur des forts gradients délimitant le contour de l'objet ou de motifs dans l'image. Nous appelons par la suite ces points les points de contour. Les points de contours sont suivis le long de la normale au contour [6] (voir Figure 1). Les résultats obtenus par un tel suivi peuvent être utilisés directement pour estimer une transformation 2D en minimisant une distance point-à-point [13]. Cependant, il n'y a pas de mise en correspondance possible entre les points de contour de \mathbf{I}^{t-1} et ceux obtenus après un tel processus de suivi bas-niveau dans \mathbf{I}^t . Le problème de mise en correspondance peut être résolu en remplaçant la minimisation d'une distance point-à-point par une minimisation d'une distance point-à-contour. La transformation 2D estimée est alors telle que :

$$\hat{\mu}_t = \underset{\mu_t}{\operatorname{argmin}} \sum_i \|d_{\perp}(\mathcal{C}_{\mu_t}, \mathbf{x}_t^i)\|^2 \quad (11)$$

où $\mathcal{C}_{\mu_t} = \Psi_{\mu_t}(\mathcal{C}_{\hat{\mu}_{t-1}})$ est le contour estimé à partir des paramètres courants de la transformation 2D μ_t , i.e. le contour défini par les points $\mathbf{x}_{\hat{\mu}_t}^i = \Psi_{\mu_t}(\mathbf{x}_{\hat{\mu}_{t-1}}^i)$ avec $\mathbf{x}_{\hat{\mu}_{t-1}}^i$ appartenant au contour $\mathcal{C}_{\hat{\mu}_{t-1}}$ estimé dans l'image précédente. \mathbf{x}_t^i sont les points obtenus par le processus de suivi bas-niveau dans \mathbf{I}^t (voir Figure 1) et $d_{\perp}(\mathcal{C}, \mathbf{x})$ est la notation utilisée pour la distance orthogonale entre le

contour \mathcal{C} et le point \mathbf{x} . Le critère (11) signifie que la transformation 2D est correctement estimée si le contour courant estimé avec les paramètres courants μ_t repose sur les points de contour extraits dans l'image courante. En se rapportant à (4), $\mathbf{m}_{\mu_t} = d_{\perp}(\mathcal{C}_{\mu_t}, \mathbf{x}_t^i)$ et $\mathbf{m}_{\mu_t}^* = d_{\perp}(\mathcal{C}_{\mu_t}^*, \mathbf{x}_t^i)$ qui est évidemment égal à zéro puisque \mathbf{x}_t^i se trouvent sur le contour $\mathcal{C}_{\mu_t}^*$ qui représente le contour de l'objet dans l'image \mathbf{I}^t . $\mathcal{C}_{\mu_t}^*$ n'a donc jamais besoin d'être calculé.

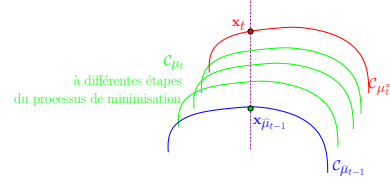


FIG. 1 – Processus de suivi le long de la normale au contour

Le calcul de \mathbf{m}_{μ_t} dépend du modèle de contour utilisé. Il est détaillé dans les deux paragraphes suivants pour le cas d'une ligne puis d'une B-Spline. La forme analytique de $\mathbf{J}_{\mathbf{m}_{\mu_t}}$ n'est pas détaillée entièrement par manque de place. Elle est obtenue dans chacun des cas en considérant que \mathbf{m}_{μ_t} peut s'exprimer comme une fonction des paramètres ϵ_j du contour. Ces dernières variables dépendent de μ_t et des paramètres initiaux du contour. On dérive alors la forme analytique de la matrice Jacobienne comme il suit :

$$\mathbf{J}_{\mathbf{m}_{\mu_t}^i} = \sum_j \frac{\partial \mathbf{m}_{\mu_t}^i}{\partial \epsilon_j} \frac{\partial \epsilon_j}{\partial \mu_t} \quad (12)$$

Suivi basé contour pour un contour polygonal. Le contour de l'objet est dans ce cas-ci modélisé par un ensemble de lignes l^j . Dans \mathbf{I}^t , le contour courant est défini par $l_{\mu_t}^j$ telles que :

$$l_{\mu_t}^j \cdot \mathbf{x}_{\mu_t}^{hi} = 0 \quad (13)$$

avec $\mathbf{x}_{\mu_t}^{hi} = (x_{\mu_t}^i, y_{\mu_t}^i, 1)^T$ les points de contours en coordonnées homogènes et $l_{\mu_t}^j = (a_{\mu_t}^j, b_{\mu_t}^j, c_{\mu_t}^j)^T$ les paramètres normalisés des droites estimés à partir des paramètres courants de la transformation 2D. La distance \mathbf{m}_{μ_t} est donnée par :

$$\mathbf{m}_{\mu_t}^i = d_{\perp}(l_{\mu_t}^j, \mathbf{x}_t^i) = l_{\mu_t}^j \cdot \mathbf{x}_t^{hi} \quad (14)$$

Les coefficients $a_{\mu_t}^j$, $b_{\mu_t}^j$ et $c_{\mu_t}^j$ sont des variables dépendant de μ_t . La forme analytique de la matrice Jacobienne peut être donc calculée à partir de (13) en utilisant (1) et (12).

Suivi basé contour pour un contour défini par une B-spline. Pour un tel contour, les points de contour $\mathbf{x}_{\mu_t}^i$ reposent sur une courbe $C_{\mu_t}(s) = (x_{\mu_t}(s), y_{\mu_t}(s))^T$ définie par :

$$C_{\mu_t}(s) : \begin{cases} x_{\mu_t}(s) &= \sum_j \alpha_{\mu_t}^j N_j(s) \\ y_{\mu_t}(s) &= \sum_j \beta_{\mu_t}^j N_j(s) \end{cases} \quad (15)$$

$\mathbf{Q}_{\mu_t}^j = (\alpha_{\mu_t}^j, \beta_{\mu_t}^j)^T$ sont les points de contrôle de la courbe et $N_j(s)$ sont les fonctions de base de la B-Spline [4]. Ces fonctions dépendent de paramètres (degré, nombre d'intervalles, ...) qui sont constants dans le temps dans le cadre de l'approche présentée ici. La distance entre un point et la B-Spline est remplacée par la distance entre ce point et la tangente à la B-Spline. Cette tangente est obtenue par :

$$C'_{\mu_t}(s) : \begin{cases} x'_{\mu_t}(s) &= \sum_j \alpha_{\mu_t}^j N'_j(s) \\ y'_{\mu_t}(s) &= \sum_j \beta_{\mu_t}^j N'_j(s) \end{cases} \quad (16)$$

Le problème de minimisation est alors similaire au cas polygonal puisque c'est encore une distance entre un point et une droite qui est utilisée, la différence étant qu'il y a dans ce cas-ci autant de lignes que de points. Cependant, il reste un problème à résoudre. Comme les B-Splines ne sont pas invariants à des transformations perspectives (*i.e.* les points de contrôle courants ne peuvent pas être obtenus directement à partir de ceux de la première image et des paramètres courants de la transformations 2D), les points de contrôle courants sont calculés par une estimation aux moindres carrés très coûteuse. Pour réduire le temps de calcul, le cadre décrit pour un contour délimité par une B-Spline a été implémenté en utilisant les NURBS (Non Uniform Rational B-Spline) [27] qui sont elles invariants à de telles transformations. La courbe $C_{\mu_t}(s) = (x_{\mu_t}(s), y_{\mu_t}(s))^T$ est alors définie par :

$$C_{\mu_t}(s) : \begin{cases} x_{\mu_t}(s) &= \frac{\sum_j \alpha_{\mu_t}^j w_{\mu_t}^j N_j(s)}{\sum_j w_{\mu_t}^j N_j(s)} \\ y_{\mu_t}(s) &= \frac{\sum_j \beta_{\mu_t}^j w_{\mu_t}^j N_j(s)}{\sum_j w_{\mu_t}^j N_j(s)} \end{cases} \quad (17)$$

où $\mathbf{Q}_{\mu_t}^j = (\alpha_{\mu_t}^j, \beta_{\mu_t}^j)^T$ sont les points de contrôle et $N_j(s)$ sont les fonctions de base de la courbe. $w_{\mu_t}^j$ est un poids associé au point de contrôle $\mathbf{Q}_{\mu_t}^j$. Il peut être interprété comme la troisième coordonnée homogène de ce point de contrôle, ce qui permet de traiter facilement les homographies [27].

Le suivi basé contour a été presque entièrement détaillé pour deux types de primitives géométriques qui couvrent une large gamme de structures planaires. Pour renforcer la robustesse du suivi basé contour par rapport aux variations du mouvement, lors du processus bas-niveau de suivi de points de contour, l'intervalle de recherche le long de la normale au contour croît quand le déplacement de l'objet dans l'image augmente et décroît quand celui-ci diminue. Le résultat obtenu par un tel algorithme est robuste aux changements globaux d'illumination et est précis quand l'objet suivi n'est pas texturé. Cependant, il nécessite une bonne initialisation et est sensible à un environnement texturé et aux déplacements importants.

2.3 Estimation d'une transformation 2D basée apparence

Le motif suivi est un sous-échantillonnage des niveaux de gris de la projection de l'objet dans l'image. La valeur de

référence de ce sous-échantillon dans \mathbf{I}^t est représentée par le vecteur $\mathbf{m}_{\mu_t}^*$ mais n'est pas mesurable directement dans l'image. L'hypothèse faible d'illumination constante donne $\mathbf{m}_{\mu_t}^* = \mathbf{m}_{\hat{\mu}_{t-1}}$ mais cette méthode entraîne généralement une dérive du motif. Par conséquent, c'est l'hypothèse forte d'illumination constante qui est utilisée ici, donnant $\mathbf{m}_{\mu_t}^* = \mathbf{m}_{\hat{\mu}_0} = \mathbf{m}_{\mu_0}^*$ où $\mathbf{m}_{\mu_0}^*$ est le motif sous-échantillonné dans la première image. La transformation 2D à estimer est alors :

$$\hat{\mu}_t = \operatorname{argmin}_{\mu_t} \left\| \underbrace{\mathbf{I}_{\mu_t}^t}_{\mathbf{m}_{\mu_t}} - \underbrace{\mathbf{I}_{\mu_0}^0}_{\mathbf{m}_{\mu_0}^*} \right\|^2 \quad (18)$$

où $\mathbf{I}_{\mu_t}^t = (I^t(x_{\mu_t}^1), \dots, I^t(x_{\mu_t}^N))^T$ contient les niveaux de gris courants sous-échantillonnés dans \mathbf{I}^t aux points $x_{\mu_t}^j = \Psi_{\mu_t}(x_{\hat{\mu}_{t-1}}^j)$. $\mathbf{I}_{\mu_0}^0$ contient les niveaux de gris sous-échantillonnés dans la première image aux points $x_{\mu_0}^j$. Les points x^j sont appelés les points de texture. La matrice Jacobienne de $\mathbf{m}_{\mu_t}^j$ est [14] :

$$\mathbf{J}_{\mathbf{m}_{\mu_t}^j} = \frac{\partial I^t(x_{\mu_t}^j)}{\partial \mu_t} = \nabla \mathbf{I}^t(\Psi_{\mu_t}(x_{\hat{\mu}_{t-1}}^j))^T \frac{\partial \Psi_{\mu_t}(x_{\hat{\mu}_{t-1}}^j)}{\partial \mu_t} \quad (19)$$

où $\nabla \mathbf{I}^t(x)$ est le gradient spatial de \mathbf{I}^t au point x . On obtient facilement $\partial \Psi_{\mu_t}(x_{\hat{\mu}_{t-1}}^j) / \partial \mu_t$ (voir [14] pour la dérivation complète et une simplification du temps de calcul).

L'estimation de la transformation 2D basée texture présentée dans cet article est proche de [7] dans le sens où c'est une homographie qui est estimée en minimisant aux moindres carrés la différence entre les motifs initial et le courant. Cependant, la matrice jacobienne n'est pas la même: dans [7], le dernier terme de l'homographie est fixé à 1 et la dérivation se fait par rapport aux coordonnées homogènes du point et non par rapport aux coordonnées dans l'image. De plus, la méthode présentée dans ce paragraphe a été modifiée pour intégrer des résultats bien connus dans le domaine de l'estimation du mouvement. Premièrement, dans ce genre de suivi, le sous-échantillonnage initial est souvent un sous-échantillonnage régulier. Or, certains points du motif apportent plus d'informations que d'autres comme expliqué dans [29]. Les valeurs propres de la matrice d'autocorrélation du signal R associée à ces points, définie par :

$$R = \begin{pmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{pmatrix} \quad (20)$$

sont élevées.

De tels points sont situés sur de forts gradients spatiaux, ce qui entraîne une meilleure estimation de la transformation 2D tandis que des points situés sur une surface plutôt uniforme ne permettent pas une estimation précise du mouvement. Pour une bonne estimation du mouvement, il faut extraire ces points d'intérêts sur la totalité de la projection

du motif, en essayant de maintenir une distance minimale entre les points sélectionnés.

Deuxièmement, l'exploitation de tels points dans le suivi implique quelques changements dans le processus de minimisation robuste. En effet, ces points étant situés sur un fort gradient spatial, un petit mouvement de l'objet dans l'image peut impliquer une plus forte augmentation du changement d'intensité pour certains points que pour d'autres. Pour éviter l'élimination des points les plus intéressants du motif par les M-estimateurs, le vecteur suivant e' est utilisé à la place du vecteur e comme vecteur de résidu pour le calcul des poids :

$$e' = \left(\dots, \frac{I^t(x_{\mu_t}^i) - I^0(x_{\mu_0}^i)}{\|\nabla I^0(x_{\mu_0}^i)\|}, \dots \right)^T \quad (21)$$

Les résultats obtenus avec un tel suivi sont robustes aux larges mouvements dans l'image et aux occultations. L'évolution des paramètres de la transformation 2D dans le temps est assez lisse, ceci étant dû au fait qu'en général, la texture est assez bien répartie sur tout l'objet. D'un autre côté, les performances sont dégradées si l'objet n'est pas "bien texturé", la précision du suivi étant alors moins bonne.

2.4 Estimation d'une transformation 2D basée sur un critère hybride

Comme il a été dit dans les paragraphes précédents, les résultats obtenus avec le suivi basé contour et le suivi basé texture sont de natures complémentaires avec chacun des propriétés intéressantes : efficacité et robustesse aux changements globaux d'illumination pour le premier, robustesse aux occultations et trajectoire lisse pour le second. Nous avons formulé ces deux algorithmes dans un formalisme similaire pour pouvoir les fusionner de manière assez directe, pour obtenir de meilleurs résultats en combinant les avantages des deux méthodes. L'approche résultant de cette fusion est décrit ci-dessous.

Le cadre général a été présenté au début de cet article indépendamment de la nature des primitives visuelles exploitées. Ceci permet de mélanger le suivi basé contour avec celui basé texture. S'il y a N_c points de contour et N_t points de texture, le vecteur *template* \mathbf{m} contenant les deux ensembles sera de taille $N = N_c + N_t$:

$$\mathbf{m}_\mu = (m_{\mu}^1, \dots, m_{\mu}^{N_c}, m_{\mu}^{N_c+1}, \dots, m_{\mu}^{N_c+N_t})^T \quad (22)$$

où $(m_{\mu}^i)_{i \leq N_c}$ est la distance point-à-contour associée au i ème point de contour et $(m_{\mu}^i)_{i=N_c+j}$ est le niveau de gris observé au j ème point de texture. (5) peut alors être appliquée pour estimer la transformation 2D en utilisant soit (12) s'il s'agit d'un point de contour, soit (19).

Cependant, l'erreur mesurée sur les points de texture (une différence d'intensité) étant d'un ordre de grandeur plus important que celle mesurée sur les points de contour (une distance point-à-contour), il est nécessaire d'effectuer une normalisation pour prendre en compte de façon équitable

les informations apportées par chaque type de primitives. Celle-ci est intégrée dans le calcul des poids qui deviennent alors :

$$w'_i = \begin{cases} \frac{w_i}{\max_c(\text{error})} & \text{if } i \leq N_c \\ \frac{w_i}{\max_t(\text{error})} & \text{if } i > N_c \end{cases} \quad (23)$$

où $\max_c(\text{error})$ (resp. $\max_t(\text{error})$) est le maximum des valeurs absolues des distances point-à-contour (resp. différences d'intensités) et w_i est le poids calculé par les M-estimateurs. De plus, il est possible de rajouter une mesure de confiance sur l'intérêt qui doit être accordé à chaque primitive. Pour les deux types de primitives, un fort gradient spatial étant une propriété intéressante, les poids finaux sont :

$$w'_i = \begin{cases} \frac{w_i \cdot \text{gradient}(i)}{\max_c(\text{error}) \cdot \max_c(\text{gradient})} & \text{if } i \leq N_c \\ \frac{w_i \cdot \text{gradient}(i)}{\max_t(\text{error}) \cdot \max_t(\text{gradient})} & \text{if } i > N_c \end{cases} \quad (24)$$

où $\max_c(\text{gradient})$ (resp. $\max_t(\text{gradient})$) représente le maximum des gradients spatiaux observés en un point de contour. (resp. point de texture) et $\text{gradient}(i)$ le gradient spatial observé au i ème point.

Fusionner les deux algorithmes de base au lieu de les enchaîner apporte des améliorations essentielles. Les expériences nous ont montré qu'inclure les deux types d'information non seulement améliore l'estimation du mouvement apparent mais élargit l'ensemble des objets pouvant être suivis. En outre, la robustesse aux mouvements importants est accrue. De plus, grâce à l'intégration des M-estimateurs, le suivi hybride est capable de gérer l'échec d'un des deux suivis élémentaires (par exemple des effets de spéularité sur une partie de l'objet pour le suivi basé apparence). Il est également important de souligner que l'échec ou les inconvénients d'un suivi élémentaire sont dûs aux données observées insuffisantes voire inexactes, menant à un minimum local plus ou moins erroné. La combinaison de deux suivis se basant sur des données différentes permet de passer ces minima locaux pour arriver à la solution qui correspond à un minimum correct pour les deux suivis.

3 Résultats expérimentaux

Les deux sous-sections suivantes présentent les résultats obtenus avec le suivi hybride sur des séquences d'images où le contour de l'objet est polygonal et les deux suivantes sur des séquences d'images où le contour est modélisé par une NURBS. Les M-estimateurs sont utilisés pour chacune des expériences puisque le contenu de la séquence vidéo n'est pas supposé connu. Ce suivi hybride a également été testé lors d'expériences d'asservissement visuel 2D dont un exemple clôt cette section résultat.

Les trois suivis décrits dans cet article ont été testés. La même quantité de données est exploitée pour chacun

d'entre eux: si $2n$ primitives sont utilisées pour un suivi basé sur un seul type d'information, alors n primitives de chaque type le sont pour le suivi hybride.

Un algorithme échoue si le contour estimé de l'objet ne correspond absolument pas au contour observé dans l'image. Lorsque plusieurs algorithmes réussissent à suivre l'objet, la comparaison se fait en calculant la différence entre le motif courant et le motif initial (*i.e.* la différence donnée par $\| \mathbf{I}_{\mu_t}^t - \mathbf{I}_{\mu_0}^0 \|$) pour chacun. Cette différence est calculée sur le motif en entier et pas seulement sur le sous-échantillonnage utilisé éventuellement pour l'estimation de la transformation 2D. Plus cette mesure est petite, plus l'estimation des paramètres de la transformation 2D est exacte. Si des occultations ont lieu, il faut noter que cette différence va augmenter puisqu'une partie du motif comparé est cachée. Cependant, si le suivi est correct, cette erreur va monter de manière similaire quelque soit l'algorithme utilisé.

Pour chaque expérience, au moins l'un des deux suivis basé sur un seul type d'information échoue alors que le second réussi éventuellement à suivre l'objet. Par contre, en ce qui concerne le suivi hybride, il réussit dans chacun des cas et il donne une meilleure estimation du mouvement si l'un des deux autres est correct. Les points choisis pour l'estimation du mouvement sont donnés dans la première image du suivi par les points verts (ou rouges si détectés comme outliers dès la première image), sauf pour la deuxième expérience par manque de visibilité. La position de l'objet est représentée par son contour en vert (estimé en appliquant le mouvement courant au contour initial) dans chaque image.

3.1 La séquence "Café terrasse la nuit" de Van Gogh

Dans cette première séquence (48 images), les déplacements inter-images sont importants. Ils peuvent atteindre 14 pixels comme montré sur la Figure 3a. Les images initiale et finale sont données pour chacune des méthodes dans la Figure 2. Le suivi basé texture perd l'objet rapidement tandis que celui basé contour donne d'assez bons résultats. Cependant, les contours ne sont pas toujours correctement recalés sur ceux observés dans l'image (voir un exemple sur la Figure 4). La seule approche donnant un suivi précis est la méthode basée sur le critère hybride. La Figure 3b présente la courbe de l'évolution de la différence entre le motif courant et le motif initial pour chacune des méthodes. Les erreurs les plus petites sont en général celles obtenues en utilisant le suivi hybride dont le comportement est plus constant que le suivi basé contour. Pour cette séquence, 470 points sont utilisés pour chacun des algorithmes. Le suivi hybride tourne à une fréquence moyenne de 13 Hz.

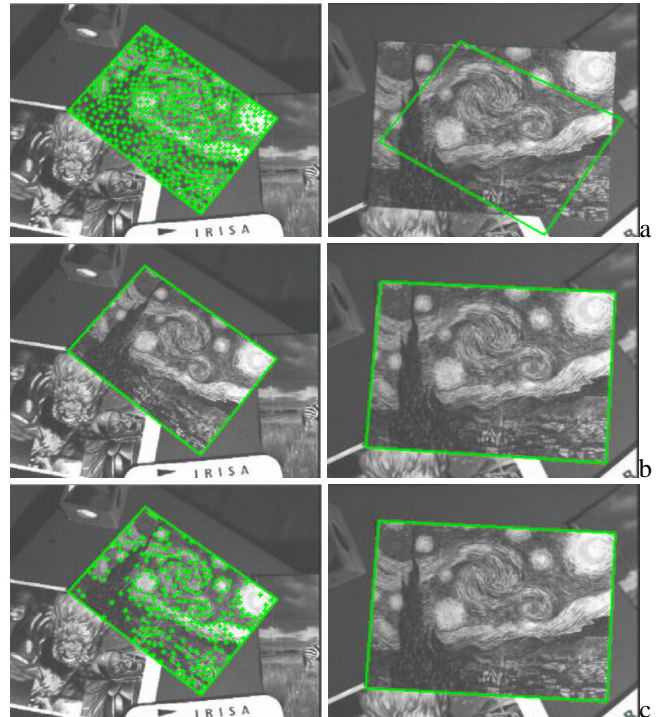


FIG. 2 – Séquence Van Gogh: suivi de contour polygonal. Images initiale et finale pour: a) suivi basé texture, b) suivi basé contour, c) suivi hybride

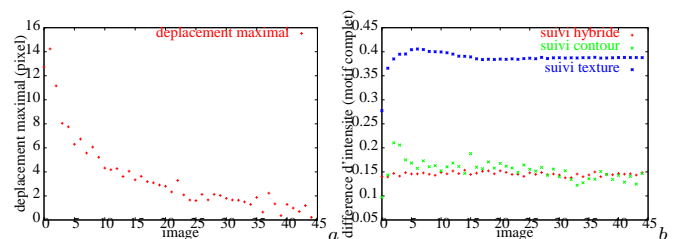


FIG. 3 – Séquence Van Gogh: suivi de contour polygonal. a) Evolution du maximum du mouvement apparent entre deux images successives (en pixel). b) Différence d'intensité entre le motif complet courant et le motif complet initial pour: bleu) suivi basé texture, vert) suivi basé contour, rouge) suivi hybride

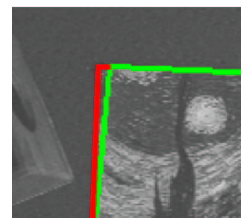


FIG. 4 – Séquence Van Gogh. Détail de la dernière image de la Figure 2b (sur le suivi basé contour): la ligne du côté gauche n'est pas précisément estimée (lignes vertes). Le contour observé dans l'image est dessiné en rouge.

3.2 La séquence du tapis de souris

Dans cette séquence composée de 600 images, l'environnement est fortement texturé. Les images initiale, intermédiaire et finale sont données pour chacune des méthodes dans la Figure 6. Comme l'on peut s'y attendre dans une telle situation, le suivi basé contour échoue à cause des contours assez ambigus. Celui basé texture et le suivi hybride réussissent à suivre l'objet de manière identique visuellement. La courbe de l'évolution de la différence entre le motif courant et le motif initial est donnée pour chacune des méthodes dans la Figure 5. Bien que le suivi basé texture repose exclusivement sur la minimisation d'une différence d'intensité, l'erreur sur le motif en entier est plus faible lorsque le suivi hybride est utilisé, ce qui illustre bien l'importance et la complémentarité des informations apportées par chaque type de primitives. 340 points sont exploités pour estimer la transformation 2D. Le suivi hybride tourne à une fréquence moyenne de 16 Hz.

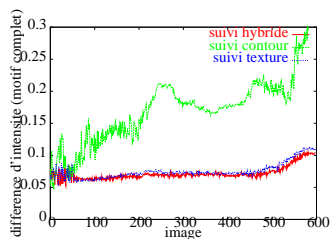


FIG. 5 – Séquence du tapis de souris: suivi de contour polygonal. Différence entre le motif courant et le motif initial pour: bleu) suivi basé texture, vert) suivi basé contour, rouge) suivi hybride

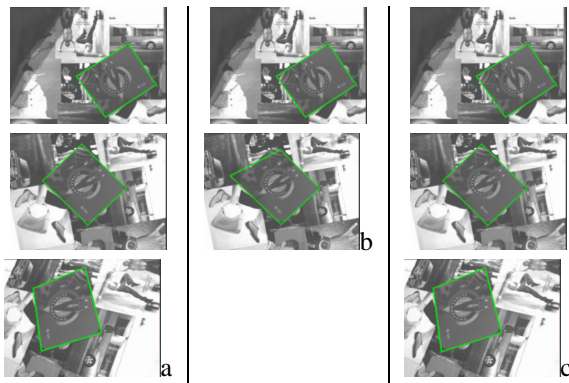


FIG. 6 – Séquence du tapis de souris: suivi de contour polygonal. Images initiale, intermédiaire et finale pour: a) suivi basé texture, b) suivi basé contour, c) suivi hybride

3.3 La séquence de la “pomme”

L'objet suivi dans cette séquence de plus de 140 images est une photo de pomme. La difficulté de cette expérience est d'obtenir un contour précis malgré l'arrière-plan texturé et l'ombre de l'image qui se confond facilement avec l'objet. Cette expérience illustre bien elle-aussi la complémentarité

des deux suivis de base. En effet, le suivi hybride réussit à suivre l'objet alors que les deux autres échouent à cette tâche. Les images initiale et finale sont données pour chacune des méthodes dans la Figure 7. 450 points ont été utilisés dans cette expérience.

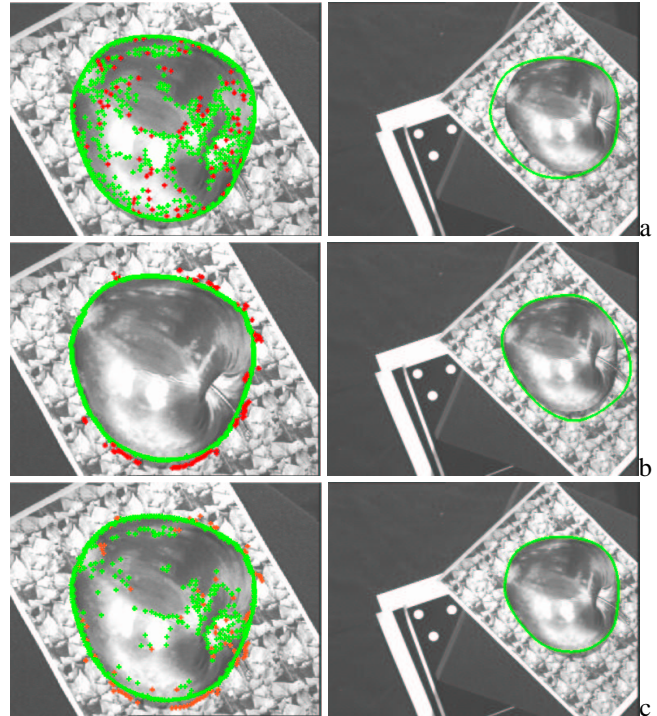


FIG. 7 – Séquence de la pomme: suivi de NURBS. Images initiale et finale pour: a) suivi basé texture, b) suivi basé contour, c) suivi hybride

3.4 La séquence de l'image du vase

Cette séquence de plus de 400 images comporte des occultations. Les images initiale, intermédiaire et finale sont données pour chacune des méthodes dans la Figure 8. Le suivi basé contour se perd lors de la seconde occultation à cause de la texture présente et du contour de l'objet occultant. Comme observé pour la deuxième séquence, bien que le suivi basé texture et le suivi hybride réussissent tous les deux à suivre l'objet, ce dernier apparaît être celui qui estime le mieux les paramètres de la transformation 2D lorsque l'on regarde les courbes de l'évolution de la différence entre le motif courant et le motif initial donnée pour chacune des méthodes dans la Figure 9. Rappelons que les deux pics sont dus aux occultations. 440 points sont exploités pour estimer la transformation 2D pour chacune des approches. Le suivi hybride tourne en moyenne à 15 Hz.

3.5 Expérience d'asservissement visuel

Le suivi hybride présenté dans cet article a été introduit avec succès dans plusieurs expériences d'asservissement visuel [12, 17]. Lors de ces expériences, les primitives

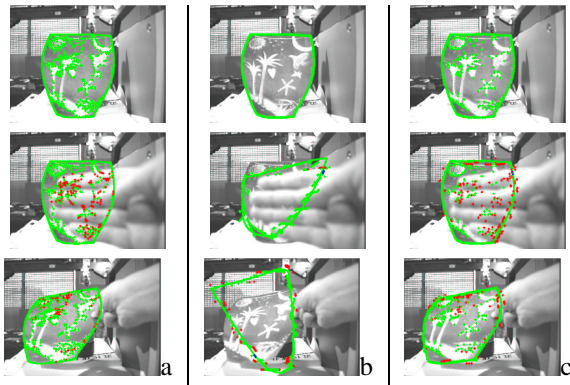


FIG. 8 – Séquence de l'image du vase: suivi de NURBS. Images initiale, intermédiaire et finale pour: a) suivi basé texture, b) suivi basé contour, c) suivi hybride

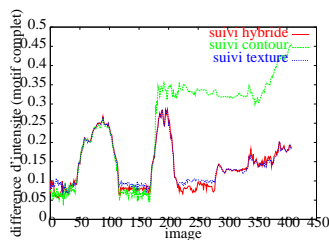


FIG. 9 – Séquence de l'image du vase: suivi de NURBS. Différence entre le motif courant et le motif initial pour: bleu) suivi basé texture, vert) suivi basé contour, rouge) suivi hybride

visuelles exploitées pour effectuer ces tâches (différentes de celles utilisées pour l'estimation de la transformation 2D) étaient les moments image de l'objet [8].

La Figure 10 présente un exemple de positionnement d'une caméra par asservissement visuel. La position désirée de l'objet dans l'image est acquise puis la caméra embarquée au bout du bras du robot à 6 degrés de liberté se déplace d'une position initiale jusqu'à la position désirée de cette caméra en minimisant l'erreur entre la position courante de l'objet dans l'image (rectangle vert) et celle désirée (rectangle rouge). Le processus de suivi de l'objet est par conséquent une étape importante dans la boucle d'asservissement visuel d'autant plus que l'utilisation de moments d'inertie comme primitives visuelles requiert une estimation précise du déplacement inter-image. Comme cela est visible sur les images, des occultations ont lieu. Les données aberrantes sont représentées par des croix rouges, les points de contour cachés par des croix bleues et les données estimées correctes par des croix vertes. La dernière image est celle obtenue lorsque la caméra atteint la position désirée.

La Figure 11 présente les positions désirée et finale de la caméra. La différence est assez faible, le positionnement de la caméra est effectué assez précisément.

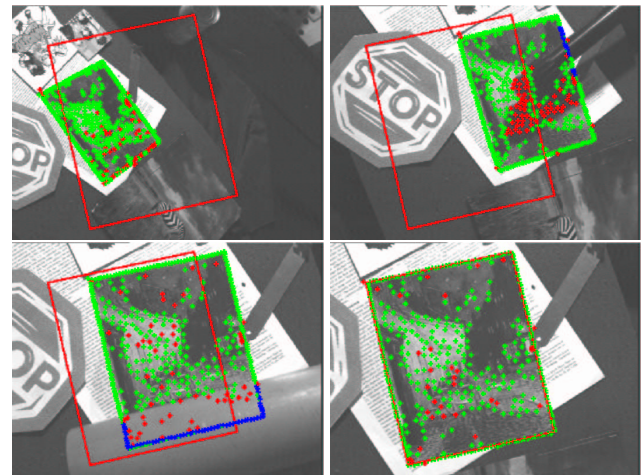


FIG. 10 – Expérience d'asservissement visuel. Rectangle rouge: position désirée de l'objet dans l'image, rectangle vert, sa position courante

Axes	t_x	t_y	t_z	r_x	r_y	r_z
Pose désirée	403.3	-51.3	300	17.8	0	5.2
Pose finale	413.8	-48.9	300.2	18	-0.2	4

FIG. 11 – Expérience d'asservissement visuel. Positions désirée et finale de la caméra en utilisant le suivi hybride (t_x, t_y et t_z en mm , r_x, r_y et r_z en degrés)

4 Conclusion

Un algorithme fiable a été présenté dans cet article. Il utilise un *template* hybride de l'objet à suivre qui regroupe les informations les plus pertinentes de son motif et d'un sous-échantillonnage régulier de son contour. En fusionnant l'estimation de mouvement apparent des points de texture et de contour, l'approche proposée permet un suivi précis d'objets texturés, même dans un environnement texturé. Elle est basée sur une minimisation itérative qui gère correctement les données aberrantes grâce à l'introduction de M-estimateurs. Le processus de minimisation est par conséquent robuste aux occultations partielles et au bruit.

Différents cas ont été présentés dans la partie résultat qui mettent bien en avant la robustesse de l'algorithme vis-à-vis de la nature de la situation ou de l'objet. Même si l'information sur le mouvement est essentiellement fiable pour un seul type de primitives, le suivi hybride permet de suivre correctement des objets dans un environnement texturé et cela assez rapidement pour être inséré dans des applications robotiques.

La transformation 2D présentée dans cet article est valable uniquement pour des objets plans. Il n'existe pas de transformation 2D qui soit utilisable pour un mouvement quelconque d'un objet 3D quelconque. Par conséquent, pour gérer de tels cas, nous nous intéressons maintenant au suivi hybride d'un objet 3D en fusionnant un calcul de pose classique et l'estimation du mouvement apparent. Un tel algo-

rithme devrait être plus robuste et moins bruité qu'un algorithme de suivi 3D classique.

Références

- [1] B. Bascle, P. Bouthemy, N. Deriche, and F. Meyer. Tracking complex primitives in an image sequence. In *ICPR'94*, pages 426–431, Jerusalem, October 1994.
- [2] S. Benhimane and E. Malis. Real-time image-based tracking of planes using efficient second-order minimization. In *IROS'04*, Sendai, Japan, October 2004.
- [3] M.-O. Berger. How to track efficiently piecewise curved contours with a view to reconstructing 3D objects. In *ICPR'94*, pages 32–36, Jerusalem, October 1994.
- [4] A. Blake and M. Isard. *Active Contours*. Springer Verlag, April 1998.
- [5] S. Boukir, P. Bouthemy, F. Chaumette, and D. Juvin. A local method for contour matching and its parallel implementation. *MVA'98*, 10(5/6):321–330, April 1998.
- [6] P. Bouthemy. A maximum likelihood framework for determining moving edges. *PAMI'99*, 11(5):499–511, May 1989.
- [7] J. Buenaposada and L. Baumela. Real-time tracking and estimation of plane pose. In *ICPR'02*, volume 2, pages 697–700, Québec, Canada, August 2002.
- [8] F. Chaumette. Image moments: a general and useful set of features for visual servoing. *ITR'04*, 20(4):713–723, August 2004.
- [9] N. Chiba and T. Kanade. A tracker for broken and closely-spaced lines. In *ISPRS'98*, pages 676 – 683., Hakodate, 1998.
- [10] A. Comport, E. Marchand, and F. Chaumette. Robust model-based tracking for robot vision. In *IROS'04*, Sendai, Japan, september 2004.
- [11] T. Drummond and R. Cipolla. Real-time visual tracking of complex structures. *PAMI'92*, 27(7):932–946, July 2002.
- [12] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *ITRA'92*, 8(3):313–326, June 1992.
- [13] A.W. Fitzgibbon. Robust registration of 2d and 3d point sets. *IVC'03*, 21(12-13):1145–1153, December 2003.
- [14] G. Hager and P. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *PAMI'98*, 20(10):1025–1039, October 1998.
- [15] G. Hager and K. Toyama. The XVision system: A general-purpose substrate for portable real-time vision applications. *CVIU'98*, 69(1):23–37, January 1998.
- [16] P.-J. Huber. *Robust Statistics*. Wiley, New York, 1981.
- [17] S. Hutchinson, G. Hager, and P. Corke. A tutorial on visual servo control. *ITRA'96*, 12(5):651–670, October 1996.
- [18] M. Isard and A. Blake. Icondensation: Unifying low-level and high-level tracking in a stochastic framework. In *ECCV'98*, pages 893–908–yy, 1998.
- [19] F. Jurie and M. Dhome. Hyperplane approximation for template matching. *PAMI'02*, 24(7):996–1000, July 2002.
- [20] D. Kragic and H. Christensen. Cue integration for visual servoing. *ITRA'01*, 17(1):19–26, February 2001.
- [21] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI'81*, pages 674–679, 1981.
- [22] E. Marchand, P. Bouthemy, F. Chaumette, and V. Moreau. Robust real-time visual tracking using a 2D-3D model-based approach. In *ICCV'99*, volume 1, pages 262–268, Kerkira, Greece, September 1999.
- [23] L. Masson, F. Jurie, and M. Dhome. Contour/texture approach for visual tracking. In *SCIA'03*, volume 2749 of *Lecture Notes in Computer Science*, pages 661–668. Springer, 2003.
- [24] I. Matthews, T. Ishikawa, and S. Baker. The template update problem. *PAMI'04*, 26(6):810–815, June 2004.
- [25] H.T. Nguyen, M. Worring, and R. van den Boomgaard. Occlusion robust adaptive template tracking. In *ICCV'01*, volume 1, pages 678–683, 2001.
- [26] J.-M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *JVCIR'95*, 6(4):348–365, December 1995.
- [27] L. Piegl and W. Tiller. *The NURBS book (2nd ed.)*. Springer-Verlag New York, Inc., 1997.
- [28] A. Shahrokni, T. Drummond, and P. Fua. Texture boundary detection for real-time tracking. In *ECCV'04*, volume 2, pages 566–577, Prague, Czech Republic, May 2004.
- [29] J. Shi and C. Tomasi. Good features to track. In *CVPR'94*, pages 593–600, Seattle, Washington, June 1994.
- [30] L. Vacchetti, V. Lepetit, and P. Fua. Stable 3-d tracking in real-time using integrated context information. In *CVPR'03*, volume 2, pages 241–248, Madison, WI, June 2003.
- [31] M. Vincze. Robust tracking of ellipses at frame rate. *Pattern Recognition*, 34(2):487 – 498, February 2001.