HAL
open science

Flot de surface à partir d'indices visuels

Benjamin Petit, Antoine Letouzey, Edmond Boyer

To cite this version:

Benjamin Petit, Antoine Letouzey, Edmond Boyer. Flot de surface à partir d'indices visuels. ORASIS - Congrès des jeunes chercheurs en vision par ordinateur, INRIA Grenoble Rhône-Alpes, Jun 2011, Praz-sur-Arly, France. inria-00595247v2

HAL Id: inria-00595247

https://inria.hal.science/inria-00595247v2

Submitted on 27 Jun 2011

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.

Flot de surface à partir d'indices visuels

Benjamin Petit Antoine Letouzey Edmond Boyer

INRIA Rhône Alpes

655, avenue de l'Europe, 38 334 Montbonnot St Ismier
prenom.nom@inrialpes.fr

Résumé

Dans ce papier nous nous intéressons à l'estimation des champs de déplacement denses d'une surface non rigide, en mouvement, capturée par un système multi-caméra. La motivation vient des applications multi-caméra qui nécessitent une information de mouvement pour accomplir des tâches telles que le suivi de surface ou la segmentation. Dans cette optique, nous présentons une approche nouvelle, qui permet de calculer efficacement un champ de déplacement 3D, en utilisant des informations visuelles de bas niveau et des contraintes géométriques. La contribution principale est la proposition d'un cadre unifié qui combine des contraintes de flot pour de petits déplacements et des correspondances temporelles éparses pour les déplacements importants. Ces deux types d'informations sont fusionnés sur la surface en utilisant une contrainte de rigidité locale. Le problème se formule comme une optimisation linéaire permettant une implémentation rapide grâce à une approche variationnelle. Les expérimentations menées sur des données synthétiques et réelles démontrent les intérêts respectifs du flot et des informations éparses, ainsi que leur efficacité conjointe pour calculer les déplacements d'une surface de manière robuste.

Mots Clef

Surface, flot, déplacements 3D.

Abstract

In this paper we study the estimation of dense 3D motion fields over non-rigidly moving surface observed by multi-camera systems. The motivation arises from multi-camera applications that require motion information in order to perform tasks such as surface tracking or segmentation. To this aim, we present a novel framework that allows to to efficiently compute dense 3D displacement fields using low level visual cues and geometric constraints. The main contribution is a unified framework that combines flow constraints for small displacements with temporal feature constraints for large displacements and fuses them over the surface using local rigidity constraints. The resulting linear optimization problem allows for variational solutions and fast implementations. Experiments conducted on synthetic and real data demonstrate the respective interests of flow and feature constraints as well as their efficiency to provide robust surface motion cues when combined.

Keywords

Surface, flow, 3D motion.

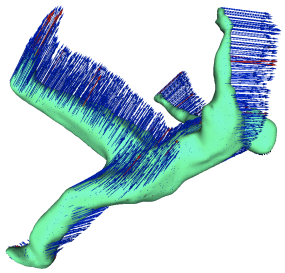


FIG. 1: Exemple de flot de scène dense (en bleu) calculé à partir de correspondances de points d'intérêts 2D et 3D (en rouge) et de flot de normal dense. Les figures de ce papier sont plus parlantes en couleur.

1 Motivations et problématique

La recherche d'une information de mouvement dense est une étape intermédiaire fondamentale de la chaîne de traitement d'images, sur laquelle peuvent être développées des applications de plus haut niveau comme, par exemple, du suivi de maillage ou de la segmentation, qui elles mêmes peuvent servir pour du transfert ou de la reconnaissance d'actions. Pour cela, l'observation des pixels, issus des images, fournit des informations utiles sur le mouvement, à travers les variations temporelles de la fonction d'intensité. Dans une configuration mono-caméra, ces variations permettent d'estimer des champs de vitesse 2D denses dans l'image : le *flot optique*. L'estimation du flot optique a été un sujet d'intérêt dans la communauté de la vision par ordinateur ces dernières dizaines d'années et de multiples méthodes ont été proposées [1, 6, 10]

Dans le cas d'un système de caméras, l'intégration depuis les différents points de vue permet de considérer le

mouvement des points 3D de la surface observée et d'estimer le champ de vecteur de déplacement 3D : le *flot de scène* [18, 12]. Pourtant, autant en 2D qu'en 3D, l'information de mouvement ne peut pas être déterminée indépendamment pour chaque point avec pour seule information la variation de la fonction d'intensité : une contrainte additionnelle doit-être introduite, par exemple, une hypothèse de continuité du champ de mouvement. De plus, du fait de l'approximation des dérivées par la méthode des différences finies, l'estimation du flot est connue pour être limitée à de petits déplacements. Bien que plusieurs approches en 2D aient été proposées pour faire face à ces limitations [20], moins d'efforts ont été consacrés au cas de la 3D.

Dans ce travail, nous avons étudié la façon d'intégrer, de manière efficace, diverses contraintes pour estimer des informations de mouvements denses instantanés sur des surfaces 3D, à partir des variations temporelles de la fonction d'intensité issue de plusieurs images. Notre motivation première a été de fournir des indices de mouvement robuste qui peuvent être directement utilisés par une application interactive, ou qui peuvent être introduits dans des applications plus avancées comme le suivi de surface ou la segmentation. Bien que notre but ait été d'intégrer le calcul des champs de vitesse avec notre application de reconstruction 3D, l'approche n'est pas limitée à un scénario spécifique et fonctionne pour toute application qui peut bénéficier d'une information de mouvement de bas niveau.

La plupart des approches existantes qui estiment le flot de scène font l'hypothèse des petits déplacements entre les instants de temps pour lesquels les approximations aux différences finies des dérivés temporelles sont valides. Cependant, cette hypothèse est souvent incorrecte avec les systèmes d'acquisition actuels et des objets réels en mouvement. En effet, l'amplitude des mouvements observés et la fréquence d'acquisition utilisée ne permettent pas d'effectuer cette hypothèse dans tous les cas.

Dans ce papier, nous avons construit une méthode unifiée permettant de lier de manière cohérente les contraintes visuelles, issues des images consécutives temporellement, avec des contraintes de déformation de surface. Pour traiter les grands déplacements, nous utilisons des mises en correspondances temporelles entre les images issues d'une même caméra. Ces contraintes agissent comme des points d'ancrage pour les régions de la surface où les déplacements sont plus importants et où les informations de variation d'intensité ne sont pas utiles. Ces contraintes visuelles sont diffusées sur la surface grâce à un schéma *Laplacien* qui régularise les vecteurs de déplacements estimés entre les points voisins de la surface. Un élément clef de cette méthode est qu'elle conduit a des optimisations linéaires ce qui permettrait, à terme, une implémentation temps-réel.

2 État de l'art

Dans un article fondateur sur le flot de scène, Vedula et al. [18] explicitent la contrainte de flot normal qui lie les dérivés de la fonction d'intensité dans les images au flot de scène des points 3D de la surface. Comme mentionné précédemment, ces contraintes ne permettent pas d'estimer le flot de scène de façon indépendante à un point de la surface, des contraintes supplémentaires doivent être introduites. Au lieu d'utiliser la contrainte de flot normal, un algorithme est proposé qui estime de façon linéaire le flot de scène à partir de la géométrie 3D de la surface et du flot optique 2D. Le flot optique permet de mieux contraindre le flot de scène que le flot normal, mais son estimation est fondée sur des hypothèses de lissage qui tiennent rarement dans le plan image mais sont souvent vérifiées dans le cas de surfaces.

Dans [12], Neumann et Aloimonos introduisent un modèle de subdivision de surface qui permet d'intégrer sur la surface, les contraintes de flot normal avec des contraintes de régularisation. Néanmoins, cette solution globale suppose encore de n'être en présence que de petits mouvements et peut difficilement faire face à des cas comme ceux présentés dans nos expérimentations.

Une autre stratégie est suivie par Pons et al. [13], qui ont présenté une approche variationnelle qui optimise un critère de cohérence photométrique au lieu des contraintes de flot normal. L'intérêt est que la cohérence spatiale comme la cohérence temporelle peuvent être appliquées, mais au prix d'une optimisation coûteuse en calcul. Au contraire, notre objectif n'est pas d'optimiser la forme observée, mais de fournir une information de mouvement dense de façon efficace et rapide.

Plusieurs travaux [22, 7, 19] considèrent le cas où la structure de la scène est décrite par une carte de disparité issue d'un système stéréoscopique. Ils proposent l'estimation combinée de la disparité spatiale et temporelle du mouvement 3D. Nous considérons une situation différente dans laquelle la surface de la forme observée est connue, par exemple, un maillage obtenu en utilisant une approche multi-vues. Ceci permet une régularisation du champ de déplacement sur un domaine où les hypothèses de régularité sont vérifiées.

Il convient de mentionner également les approches récentes sur le suivi temporel de surface [16, 17, 11, 2] qui peuvent également fournir des champs de vitesse. C'est en effet une conséquence de la mise en correspondance de surfaces dans le temps. Notre but est ici différent, notre méthode ne fait aucune hypothèse sur la forme observée, seulement quelques hypothèses sur le modèle de déformation locale de la surface. Notre méthode fournit des informations bas niveau, le mouvement instantané, qui peuvent à leur tour être utilisées comme données d'entrée d'une méthode d'appariement ou de suivi de surface.

Nos contributions à l'égard des approches mentionnées sont de deux ordres :

– En suivant les travaux sur l'estimation robuste du flot optique 2D [8, 20], nous utilisons avantageusement les valeurs de déplacement robuste fournies par le suivi de points d'intérêts dans des images consécutives dans le

temps. Ces points intérêts permettent de contraindre les grands déplacements alors que les contraintes de flot de normal permettent de modéliser précisément les déplacements les plus petits.
– Une résolution linéaire combine ces différentes contraintes visuelles avec un modèle de déformation de surface et permet une résolution itérative ainsi qu'un raffinement de type multi-échelle.

3 Définitions préliminaires

Notre méthode peut s'appliquer en sortie de tout système multi-caméra capable de produire un flux de surface se déplaçant de manière non rigide. Chacune de ces surfaces ayant été modélisée indépendamment de la précédente, à partir des N caméras calibrées du système en utilisant une technique de reconstruction 3D telles que [4] ou [5].

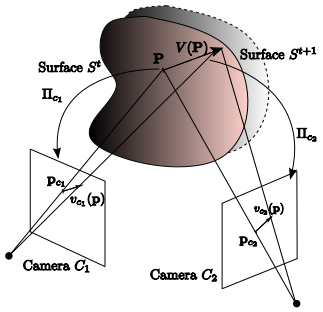


FIG. 2: Projection du flot de scène en flot optique sur les différentes images issues du système multi-caméra.

La surface au temps t est dénotée $\mathcal{S}^t \subset \mathbb{R}^3$ et associée au jeu d'images $\mathcal{I}^t = \{\mathbf{I}_c^t \mid c \in [1..N]\}$. Un point 3D $\mathbf{P}$ sur la surface est décrit par le vecteur $(x, y, z)^T \in \mathbb{R}^3$. Sa projection dans l'image $\mathbf{I}_c^t$ est le point 2D $\mathbf{p}_c$ qui a comme coordonnées $(u_c, v_c)^T \in \mathbb{R}^2$, calculées en utilisant la matrice de projection 3x4 $\mathbf{\Pi}_c : \mathbb{R}^3 \mapsto \mathbb{R}^2$ de la caméra c (voir figure 2). La région 3D de l'image correspondant à la visibilité de $\mathcal{S}^t$ dans $\mathbf{I}_c^t$ est notée $\Omega_c^t = \mathbf{\Pi}_c \mathcal{S}^t$.
Notre méthode recherche le meilleur champ de déplacement 3D de la surface entre le temps t et $t+1$, noté $V^t : \mathcal{S}^t \mapsto \mathbb{R}^3$ avec $V^t(\mathbf{P}) = \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t}\ \forall\, \mathbf{P} \in \mathcal{S}^t$. Ce champ de déplacement est contraint par :
– les données d'entrée comme le jeu d'images calibrées $\mathcal{I}^t$ et $\mathcal{I}^{t+1}$, et la surface $\mathcal{S}^{t+1}$,
– un modèle de déformation.

La projection du champ de déplacement 3D sur $\mathbf{I}_c^t$ est décrit par v_c^t. La relation entre un petit déplacement à la surface de $\mathcal{S}^t$ et son image prise par la caméra c est décrite par la matrice *Jacobienne* 2x3 $J_{\mathbf{\Pi}_c}(\mathbf{p}_c) = \frac{\partial \mathbf{p}_c}{\partial \mathbf{P}}$, tel que $v_c^t = J_{\mathbf{\Pi}_c}(\mathbf{p}_c) V^t$.
Dans les sections suivantes, nous expliciterons les contraintes visuelles et géométriques décrites par les données d'entrée et le modèle de déformation utilisé pour propager le mouvement sur la surface.

4 Contraintes visuelles

Notre méthode peut utiliser trois types de contraintes visuelles pour estimer le déplacement 3D :

1. des correspondances éparses de points d'intérêts 3D,
2. des correspondances éparses de points d'intérêts 2D,
3. des contraintes denses de flot normal dans les images.

Chacune de ces contraintes mènera à un terme dans notre fonctionnelle d'erreur qui sera présentée dans la section §5.2 et qui décrit comment le champ de déplacement estimé se rapporte aux observations. Ces contraintes n'incluent pas de cohérence photométrique spatiale ou temporelle car ces dernières impliquent des termes non linéaires dans la fonctionnelle d'erreur. Elles sont plus adaptées aux problèmes liés à l'optimisation de la forme de la surface qu'à l'estimation plus bas niveau du mouvement.

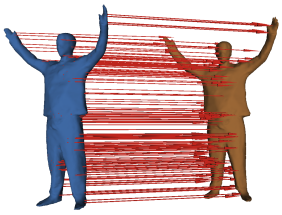
4.1 Correspondances 3D éparses

La mise en correspondance de points d'intérêts 3D permet de recueillir de l'information pour un jeu de points 3D à la surface de $\mathcal{S}^t$. Ces points 3D et leurs déplacements associés sont obtenus par la détection des points d'intérêts 3D sur $\mathcal{S}^t$ et $\mathcal{S}^{t+1}$, en leur créant un descripteur et en les appariant grâce à la comparaison de ces descripteurs.
Il existe différentes voies pour obtenir des correspondances 3D entre deux formes. Dans notre approche, nous utilisons MeshDOG pour détecter des points d'intérêts 3D et MeshHOG pour les décrire [21]. Cette méthode définit et met en correspondance les extremas locaux de n'importe quelle fonction scalaire définie sur la surface. D'autres techniques de détection et mise en correspondance peuvent être utilisées (tel que [15]), tant qu'elles récupèrent un ensemble de correspondances robustes entre les deux surfaces.
L'avantage de l'utilisation de points d'intérêts 3D est que, contrairement au flot optique (décrit à la section §4.3), ils permettent de contraindre le mouvement même lors de grands déplacements dans l'espace 3D.
On obtient un ensemble épars de déplacements 3D V_m^t pour des points 3D $\mathbf{P}_m \in \mathcal{S}^t$ (voir figure 3-(a)). Ces points forment un sous-ensemble discret de $\mathcal{S}^t$ appelé $\mathcal{S}_m^t$. La fonction d'erreur suivante décrit la proximité du champ de déplacement calculé V^t au champ de déplacement épars V_m^t :

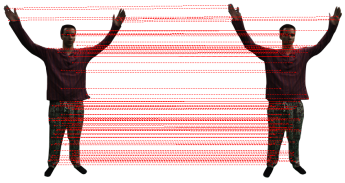
$$\mathbf{E}_{3D} = \sum_{\mathcal{S}_m^t} \|V^t - V_m^t\|^2 \ . \tag{1}$$

4.2 Correspondances 2D éparses

Dans notre approche, nous considérons des correspondances 2D éparses entre les jeux d'images $\mathcal{I}^t$ et $\mathcal{I}^{t+1}$. Comme dans le cas de la 3D, il y a différentes techniques existantes pour calculer des correspondances 2D entre une paire d'images, par exemple, SIFT, SURF ou Harris. Sans pour autant perdre en généralité, nous nous appuyons sur le détecteur et descripteur SIFT [9]. Il s'est avéré robuste



(a)



(b)

FIG. 3: Correspondances 3D éparses entre deux surfaces (a) et correspondances 2D éparses entre deux images (b).

et bien adapté dans notre cas, car invariant aux rotations et aux changements d'échelle.
Nous calculons des points d'intérêts sur les flux d'images $\mathcal{I}^t$ et $\mathcal{I}^{t+1}$. Nous mettons ensuite en correspondance les points d'intérêts venant des images $\mathbf{I}_c^t$ avec ceux de $\mathbf{I}_c^{t+1}$. Cela nous donne un jeu de déplacements 2D épars $v_{c,s}^t$ pour quelques points 2D $\mathbf{p}_{c,s} \in \Omega_c^t$ (voir figure 3-(b)). Ces point forment un sous-ensemble de Ω_c^t appelé $\Omega_{c,s}^t$. La fonction d'erreur suivante décrit la proximité du champ de déplacement 2D calculé v_c^t au champ de déplacement 2D épars $v_{c,s}^t$:

$$\mathbf{E}_{2D} = \sum_{c=1}^{N} \sum_{\Omega_{c,s}^t} \|v_c^t - v_{c,s}^t\|^2 \,,$$

ce qui est équivalent à :

$$\mathbf{E}_{2D} = \sum_{c=1}^{N} \sum_{\Omega_{c,s}^t} \|J_{\mathbf{\Pi}_c} V^t - v_{c,s}^t\|^2 \,. \tag{2}$$

Il est important de noter que des correspondances 3D peuvent être obtenues à partir des points d'intérêts 2D en re-projetant les points détectés depuis $\mathcal{I}^t$ sur la surface $\mathcal{S}^t$ et ceux de $\mathcal{I}^{t+1}$ sur la surface $\mathcal{S}^{t+1}$. Cela fournit une liste de points d'intérêts 3D qui peuvent être mis en correspondance grâce à leurs descripteurs SIFT. Au lieu de réaliser cette mise en correspondance seulement dans l'espace d'une seule image, cela permet de prendre en compte les descripteurs issus de plusieurs images. Dans ce cas, la fonctionnelle d'erreur est similaire à $\mathbf{E}_{3D}$ décrite dans l'équation (1).
Bien que les points d'intérêts 3D soient plus robustes, en particulier aux occultations, et fournissent une meilleure information sur de longues séquences, ils présentent des désavantages, par rapport aux points d'intérêts 2D, pour l'estimation du flot de scène. Ils ne sont pas robustes aux changements de topologie et sont plus demandeurs en puissance de calcul, ce qui peut être crucial dans certaines applications. De plus, la surface S^{t+1} est forcément requise ce qui peut être problématique pour des applications interactives.

4.3 Flot normal dense

Contrairement aux contraintes éparses données par les correspondances 2D et 3D, des informations denses sur V^t peuvent être obtenues en utilisant le flot optique accessible dans les images. En effet, en prenant comme hypothèse que l'illumination reste constante entre $\mathbf{p}_c^{t+1}$ et $\mathbf{p}_c^t$, la projection du même point de la surface entre deux trames consécutives, on peut définir l'équation du *flot normal* [1] comme étant :

$$\nabla I_c^t . v_c^t + \frac{\mathrm{d}I_c^t}{\mathrm{d}t} = 0 \,,$$

ou équivalent en 3D à [18] :

$$\nabla I_c^t . \left[J_{\mathbf{\Pi}_c} V^t \right] + \frac{\mathrm{d}I_c^t}{\mathrm{d}t} = 0 \,.$$

La fonction d'erreur suivante décrit comment le champ de déplacement 2D calculé v_c^t vérifie la contrainte de flot normal :

$$\mathbf{E}_{flow} = \sum_{c=1}^{N} \int_{\Omega_c^t} \|\nabla I_c^t . \left[J_{\mathbf{\Pi}_c} V^t \right] + \frac{\mathrm{d}I_c^t}{\mathrm{d}t}\|^2 \, \mathrm{d}\mathbf{p}_c \,. \tag{3}$$

Cependant, cela ne contraint les déplacements 2D que dans la direction du gradient de l'image ∇I_c^t, qui est la composante normale du flux optique. Comme il sera expliqué dans la section suivante, c'est le *problème de l'ouverture* connu en 2D et étendu à la 3D.

5 Régularisation

Les correspondances éparses 2D et 3D contraignent seulement le déplacement de la surface pour des points 3D spécifiques et pour leur re-projection dans les images. Pour trouver un champ de mouvement dense sur la surface, nous avons besoin de propager ces contraintes en utilisant un terme de régularisation.
En outre, comme mentionné précédemment, les contraintes denses de flot normal ne fournissent pas assez de contraintes pour estimer les déplacements 3D. En effet, il peut être démontré que les équations du flot normal pour des projections dans différentes images d'un même point 3D $\mathbf{P}$ contraignent de façon indépendantes V^t à $\mathbf{P}$, et ne résolvent donc que 2 degrés de liberté sur 3. Vedula et al. [18] mentionnent deux stratégies de régularisation pour faire face à cette limitation. La régularisation peut être effectuée dans les plans images en estimant les flux optiques qui fournissent des contraintes plus complètes sur le flot de scène, ou elle peut être effectuée sur la surface 3D.

Puisque nous avons connaissance de la surface 3D et que les contraintes éparses 2D et 3D doivent être également intégrées, un choix naturel dans notre contexte est de régulariser en 3D. En plus, la régularisation dans l'espace image souffre d'artefacts et d'incohérences résultant des discontinuités de profondeur et des occultations qui contredisent l'hypothèse de lissage, alors qu'une telle hypothèse se justifie sur la surface 3D.

5.1 Modèle de déformation

Les hypothèses de régularité sur les champs de déplacement 3D de la surface limitent les déformations de cette surface à un niveau local. Elles définissent ainsi un modèle de déformation de la surface, par exemple, une rigidité locale. En 2D, de nombreuses méthodes de régularisation ont été proposées pour l'estimation du flot optique, elles se répartissent en 2 grandes catégories : les régularisations locales ou globales. Elles peuvent être étendues à la 3D.
Par exemple, la méthode 2D de Lucas et Kanade, qui utilise un voisinage local, a été appliquée en 3D par Devernay et al. [3]. Toutefois, le modèle de déformation associé à la surface n'a pas de signification réelle, car les contraintes de déformation ne se propagent que localement, ce qui amène à des incohérences entre les voisins. D'autre part, la stratégie globale introduite par Horn et Schunck [6] est bien mieux adaptée à notre contexte. Bien que moins robuste au bruit que les méthodes locales telles que Lucas-Kanade, elle permet la propagation de contraintes éparses sur toute la surface. En outre, le modèle de déformation associé a prouvé son efficacité dans le domaine du graphisme [14].
L'extension du modèle de déformation d'Horn et Schunck à des points 3D est décrit par la fonction d'erreur suivante qui assure une rigidité locale du champ de mouvement :

$$\mathbf{E}_d = \int_S \|\nabla V\|^2 \mathrm{d}\mathbf{P}\,. \tag{4}$$

5.2 Minimisation de la fonctionnelle d'énergie

Nous trouvons le meilleur déplacement qui satisfait toutes les contraintes susmentionnées en minimisant la fonctionnelle d'erreur suivante :

$$\arg\min_V \left[\lambda_{3D}^2 \mathbf{E}_{3D} + \lambda_{2D}^2 \mathbf{E}_{2D} + \lambda_{flow}^2 \mathbf{E}_{flow} + \lambda_d^2 \mathbf{E}_d\right]\,, \tag{5}$$

où les différents coefficients λ sont des paramètres qui peuvent être configurés pour donner plus de poids à une contrainte particulière.
Cette fonctionnelle peut être minimisée en résolvant l'équation d'Euler-Lagrange qui lui est associée :

$$\sum_{c=1}^{N} \left[\lambda_{flow}^2 \left[\nabla I_c^t . \left[J_{\mathbf{\Pi}_c} V^t\right] + \frac{\mathrm{d}I_c^t}{\mathrm{d}t}\right] + \lambda_{2D}^2 \delta_{\Omega_{c,s}^t} J_{\mathbf{\Pi}_c}\left[V^t - V_{c,s}^t\right]\right] + \lambda_{3D}^2 \delta_{\mathcal{S}_m^t}\left[V^t - V_m^t\right] + \lambda_d^2 \nabla^2 V^t = 0\,, \tag{6}$$

où δ est le symbole de Kronecher, qui précise que les contraintes éparses ne sont définies que pour certains points 3D appartenant à $\mathcal{S}_m^t$ ou leurs projections à $\Omega_{c,s}^t$.
L'équation d'Euler-Lagrange discrétisée pour chaque point 3D $\mathbf{P}$ de la surface a la forme :

$$\mathbf{A_P} V_{\mathbf{P}} + \mathbf{b_P} - \Delta V_{\mathbf{P}} = 0\,, \tag{7}$$

où Δ est l'opérateur de Laplace-Beltrami normalisé sur la surface.
La combinaison de l'équation (7) pour tous les points 3D $\mathbf{P} \in \mathcal{S}^t$ crée un système linéaire simple de la forme :

$$\begin{bmatrix}\mathbf{L}\\ \mathbf{A}\end{bmatrix} V^t + \begin{bmatrix}\mathbf{0}\\ \mathbf{b}\end{bmatrix} = 0\,, \tag{8}$$

où $\mathbf{L}$ est la matrice laplacienne définie dans [14]. C'est un système linéaire très épars qui peut être résolu en utilisant n'importe quel *solver* épars, *Taucs*, par exemple.
Il est intéressant de noter que cette formulation revisite la méthode d'édition Laplacienne de maillage *« as-rigid-as-possible »* de la communauté du graphisme [14]. Il est connu que ce modèle de déformation ne gère pas explicitement les rotations de la surface. Bien que cela soit un problème quand on déforme la surface avec peu de contraintes, ce qui est courant dans les applications de graphisme, la densité de contrainte de flot normal dans notre cas aide à récupérer les rotations sans pour autant passer par une optimisation non-linéaire du problème.
L'équation (6) peut aussi être résolue itérativement en utilisant la méthode de Jacobi appliquée à ce large système linéaire. Dans ce cas, il est possible de résoudre ce système linéaire pour chaque point indépendamment et de répéter cette étape itérativement en utilisant la solution mise à jour des points voisins. Cette approche variationnelle permet également d'affiner les résultats dans un contexte multi-échelle.

5.3 Sélection des poids et affinage itératif

Dans l'équation (6), les paramètres λ_{2D}, λ_{3D}, λ_{flow} et λ_d indiquent le poids, respectivement, des points d'intérêts 2D et 3D, du flot normal 2D et du Laplacien. Une forte valeur indique une influence plus importante pour le terme associé.
Dans notre contexte, de manière similaire à [20] en 2D, nous faisons confiance à nos points d'intérêts pour être robustes même lors de grands déplacements et nous sommes conscients que les contraintes de flot ne sont pas fiables quand la re-projection du déplacement est plus grande que quelques pixels dans les images. En conséquence, nous proposons une méthode itérative qui effectue deux minimisations successives de la fonctionnelle d'énergie avec deux jeux de paramètres différents. Les étapes de notre algorithme sont les suivantes :

1. Nous commençons par calculer les correspondances éparses 2D et 3D entre $\mathcal{S}^t$ et $\mathcal{S}^{t+1}$ et entre $\mathcal{I}^t$ et $\mathcal{I}^{t+1}$. Nous calculons également la matrice Laplacienne $\mathbf{L}$ de notre surface discrétisée.

2. Nous résolvons l'équation (8), avec $\lambda_{flow} = 0$ et des valeurs plus importantes pour λ_{3D} et λ_{2D} que pour λ_d. Nous obtenons alors une première estimation de V^t dénotée V'^t qui récupère les grands déplacements de la surface.
3. Nous créons une surface déformée $\mathcal{S}'^t = \mathcal{S}^t + V'^t$ que nous projetons dans toutes les caméras en utilisant l'information de texture d'origine, venant de la projection de $\mathcal{I}^t$ sur $\mathcal{S}^t$. Nous obtenons alors un nouveau jeu d'images $\mathcal{I}'^t$.
4. Nous calculons alors la visibilité de la surface S'^t sur chaque caméra ainsi que les contraintes denses de flot normal entre $\mathcal{I}'^t$ et $\mathcal{I}^{t+1}$ pour chaque point visible de la surface. Nous obtenons donc plusieurs contraintes par points échantillonnés sur la surface.
5. Tout comme dans l'étape 2, nous résolvons l'équation (8) en utilisant le flot calculé dans l'étape 4 et les points d'intérêts 2D et 3D calculés précédemment dans l'étape 1. Ces derniers sont utilisés comme des points d'ancrage ayant une contrainte de déplacement nul. Pour cette étape, nous utilisons des valeurs fortes de λ_{3D} et λ_{2D} et des valeurs plus faibles pour λ_{flow} et λ_d. Nous obtenons alors le déplacement entre $\mathcal{S}'^t$ et $\mathcal{S}^{t+1}$ dénoté V''^t et donc également une version raffinée de $V^t = V'^t + V''^t$. Cette seconde minimisation permet de récupérer de plus petits déplacements, mieux contraints par les contraintes de flot.

Nous avons observé par nos résultats que, dans la pratique, notre approche peut gérer aussi bien de grands déplacements que des petits. Ceci grâce aux points d'intérêts qui gèrent bien les grands déplacements et au flot de normal qui récupère mieux les détails précis.

6 Évaluation

Pour notre évaluation nous avons utilisé aussi bien des données synthétiques que des données réelles :

1. Les données synthétiques ont été obtenues grâce à un modèle d'humain articulé, déformé au cours du temps pour créer une séquence de danse. Nous avons rendu cette séquence dans dix caméras virtuelles de résolution 1 MPixels, réparties sur une sphère autour de la danseuse. Le modèle utilisé est un maillage triangulaire avec $7K$ sommets, déformé pour générer une séquence de 200 trames.
2. Les données réelles ont été récupérées à partir de banques de données accessibles au public. La première séquence a été prise à partir de 32 caméras 2 MPixels. Les maillages, obtenus avec EPVH, comportent $\sim 10K$ sommets. Nous avons également utilisé la séquence du *flashkick* de la base de données multi-vidéo *SurfCap* [16] de l'Université de Surrey. Cette séquence a été enregistrée à partir de huit caméras 2 MPixels, et produit des maillages lisses de $\sim 140K$ sommets.

6.1 Évaluation quantitative sur des données synthétiques

Grâce à l'algorithme décrit dans la section §5.3, nous avons calculé les champs de mouvement sur la séquence synthétique de la danseuse. La figure 4-(a)-(b)-(c) montre le champ de déplacement sur une des trames de la séquence. Les flèches rouges désignent les contraintes issues des points d'intérêts 3D et de la projection des points d'intérêts 2D, alors que les bleues désignent les vecteurs du champ de déplacement dense 3D.
La figure 4-(d) montre le champ de déplacement accumulé sur plusieurs trames à partir d'une vue de dessus. Ce résultat peut être en quelque sorte comparé à celui de Varanasi et al. [17], en effet leur méthode, permet de mettre en correspondance deux maillages consécutifs dans le temps, et est capable de fournir un champ de vitesse en conséquence de cet appariement.
Comme les maillages sont cohérents dans le temps, nous avons pu obtenir la réalité terrain et donc évaluer nos résultats quantitativement. La figure 5 montre l'erreur sur l'angle et la taille de chaque vecteur de mouvement après chacune des deux étapes de régularisation de notre algorithme. Nous pouvons voir les avantages de l'utilisation des contraintes de flot normal pour affiner le champ de déplacement (voir gros plan sur la figure 6).

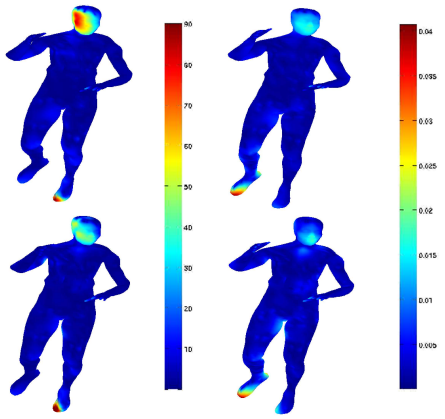


FIG. 5: Erreur sur le champ de déplacement : en angle en degré (gauche) et en norme en mètre (droite), après la première (haut) et la deuxième (bas) régularisation.

6.2 Expériences sur des données réelles

Notre première séquence réelle montre un sujet qui réalise des actions simples : il déplace ses deux mains à partir des hanches jusqu'au dessus de sa tête. Le sujet porte des vêtements amples et bien texturés ce qui permet de calculer un nombre élevé et fiable de correspondances 2D et 3D.
Les figures 7-(a)-(b)-(c) montrent le mouvement instantané récupéré en utilisant notre méthode. Notez que nous ne calculons qu'un mouvement dense sur la surface et non une déformation du maillage. Ainsi, nous n'avons pas une connectivité constante dans le temps et nous ne pouvons

(a) (b) (c) (d)

FIG. 4: Champ de déplacement sur plusieurs trames de notre séquence synthétique de danseuse, et historique du mouvement sur plusieurs trames vu du dessus (les couleurs indiquent l'ancienneté du mouvement).

(a) Première régularisation (b) Seconde régularisation

FIG. 6: Gros plan sur l'erreur en angle sur la face de la danseuse. Ces images montrent l'amélioration après la deuxième étape de régularisation qui aide pour récupérer les petits déplacements.

pas effectuer le suivi des sommets du maillage sur toute la séquence. Par conséquent, l'évaluation quantitative des données n'est pas possible, mais la visualisation des résultats est très satisfaisante. La figure 7-(d) montre le champ de déplacement accumulé sur toute la séquence.

Nous avons également calculé le champ de déplacement 3D sur la séquence du *flashkick* qui est très populaire. Dans cette séquence difficile, le sujet porte des vêtements amples avec peu d'information de texture. En outre, l'amplitude du mouvement est très élevée entre deux trames. Nous pouvons donc calculer moins de correspondances 2D et 3D. Elles sont pourtant nécessaires pour récupérer les gros déplacements.

Nous avons cependant réussi à calculer un champ de mouvement cohérent sur la plupart des trames (voir les figures 8-(a)-(b)). Sur certaines trames, notre algorithme n'a pas trouvé de points d'intérêts sur les jambes ou les pieds du danseur, le champ de mouvement calculé à partir de ces indices montre bien la bonne direction, mais pas la bonne norme des vecteurs. Le manque de contraintes visuelles pour la première estimation du champ de mouvement ne permet pas de calculer certains déplacements complètement, le déplacement restant ne peut pas être récupéré entièrement avec les contraintes de flot normal. La figure 8-(c) montre une trame problématique où le mouvement de la jambe droite du danseur n'est pas correctement calculé. Pour visualiser cette erreur, nous avons affiché les surfaces d'entrée au temps t et $t+1$ (respectivement cyan et bleu foncé), tandis que la surface déplacée avec le champ de mouvement calculé est indiquée par des points jaunes. Enfin, la figure 8-(d) montre l'historique du mouvement sur quelques trames.

7 Bilan

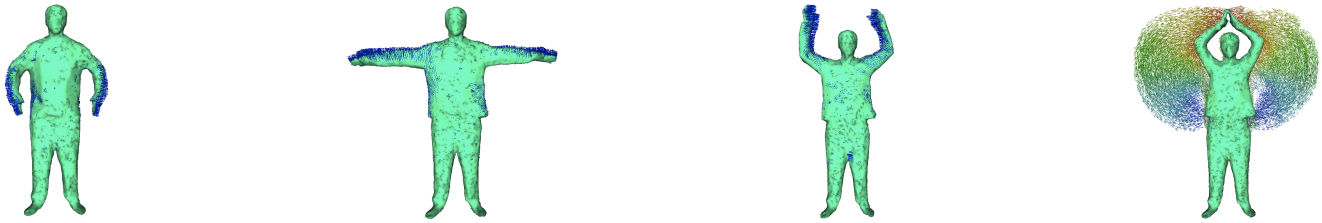
La contribution de ce travail est double : premièrement, nous avons présenté une méthode unifiée qui permet de combiner des informations photométriques pour calculer le mouvement d'une surface, d'autre part, nous avons introduit une méthode itérative qui permet de gérer de gros déplacements tout en récupérant les petits détails.

Comme le montrent les résultats, notre méthode est assez robuste. Néanmoins, nos expériences ont mis en évidence certaines faiblesses potentielles. Comme nous le pensions, en s'appuyer sur des caractéristiques visuelles impose d'avoir une bonne information de texture dans les images. Notre méthode pourrait être améliorée par l'ajout d'autres contraintes, par exemple, un critère de cohérence photométrique tel que celui utilisé par Pons et al. dans [13].

La méthode que nous proposons donne de toute façon des informations utiles et fiables sur les propriétés intrinsèques d'une séquence 4D. La connaissance du déplacement instantané peut être utilisée comme donnée d'entrée pour de nombreuses tâches en vision par ordinateur, telles que le suivi de surface, le transfert de mouvement ou la segmentation de maillage.

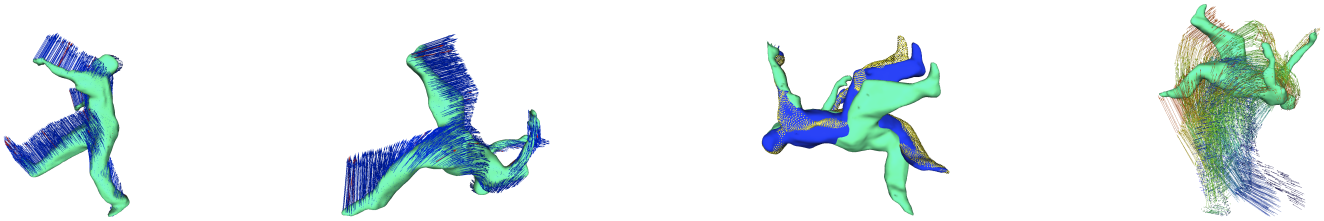
Même si nous n'avons pas mis l'accent sur les performances de calcul pour notre première implémentation, nous sommes certains que la plupart des calculs pourraient s'exécuter en parallèle. En effet, l'extraction des points d'intérêts 2D, ainsi que le calcul des contraintes de flot normal, sont indépendants par caméra. De plus, des implémentations temps-réel de SIFT et des méthodes de flux optique existent déjà. La propriété linéaire de la régularisation permet de s'attendre à une exécution en temps-réel également.

Pour le moment, les systèmes multi-caméra qui permettent de faire de la reconstruction 3D en temps-réel ne calculent pas vraiment le mouvement associé au modèle. Ces informations supplémentaires pourraient améliorer considérablement les applications interactives telles que les interactions basées sur des collisions entre le sujet reconstruit et toutes sortes d'objets virtuels. Il pourrait également être utilisé pour la reconnaissance d'action.



(a) (b) (c) (d)

FIG. 7: Champs de déplacement sur certaines trames de nos données réelles, et historique du mouvement accumulé sur toute la séquence (les couleurs indiquent l'ancienneté du mouvement).



(a) (b) (c) (d)

FIG. 8: Champs de déplacement sur plusieurs trames de la séquence du flashkick (a) et (b), mouvement partiellement calculé (c), et historique du mouvement sur toute la séquence (d) (les couleurs indiquent l'ancienneté).

Références

[1] J. Barron, D.-J. Fleet, and S. Beauchemin. Performance of Optical Flow Techniques. *International Journal of Computer Vision*, 1994. 1, 4

[2] C. Cagniart, E. Boyer, and S. Ilic. Probabilistic Deformable Surface Tracking From Multiple Videos. In *European Conference on Computer Vision*, 2010. 2

[3] F. Devernay, D. Mateus, and M. Guilbert. Multi-Camera Scene Flow by Tracking 3-D Points and Surfels. In *Computer Vision and Pattern Recognition*, 2006. 5

[4] J.-S. Franco and E. Boyer. Efficient Polyhedral Modeling from Silhouettes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008. 3

[5] Y. Furukawa and J. Ponce. Carved Visual Hulls for Image-Based Modeling. In *European Conference on Computer Vision*, 2006. 3

[6] B. Horn and B. Schunck. Determining Optical Flow. *Artificial Intelligence*, 1981. 1, 5

[7] M. Isard and J. MacCormick. Dense Motion and Disparity Estimation via Loopy Belief Propagation. In *Asian Conference on Computer Vision*, 2006. 2

[8] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. Freeman. SIFT Flow : Dense Correspondence across Different Scenes. In *European Conference on Computer Vision*, 2008. 2

[9] D. Lowe. Distinctive Image Features from Scale-invariant Keypoints. *International Journal of Computer Vision*, 2004. 3

[10] B. Lucas and T. Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In *International Joint Conference on Artificial Intelligence*, 1981. 1

[11] A. Naveed, C. Theobalt, C. Rossl, S. Thurn, and H. Seidel. Dense Correspondence Finding for Parametrization-free Animation Reconstruction from Video. In *Computer Vision and Pattern Recognition*, 2008. 2

[12] J. Neumann and Y. Aloimonos. Spatio-Temporal Stereo Using Multi-Resolution Subdivision Surfaces. *International Journal of Computer Vision*, 2002. 2

[13] J.-P. Pons, R. Keriven, and O. Faugeras. Modelling Dynamic Scenes by Registering Multi-View Image Sequences. In *Computer Vision and Pattern Recognition*, 2005. 2, 7

[14] O. Sorkine and M. Alexa. As-Rigid-As-Possible Surface Modeling. In *Eurographics Symposium on Geometry Processing*, 2007. 5

[15] J. Starck and A. Hilton. Correspondence Labeling for Wide-Timeframe Free-Form Surface Matching. In *European Conference on Computer Vision*, 2007. 3

[16] J. Starck and A. Hilton. Surface Capture for Performance-Based Animation. *IEEE Computer Graphics and Applications*, 2007. 2, 6

[17] K. Varanasi, A. Zaharescu, E. Boyer, and R. P. Horaud. Temporal Surface Tracking Using Mesh Evolution. In *European Conference on Computer Vision*, 2008. 2, 6

[18] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-Dimensional Scene Flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005. 2, 4

[19] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremeres. Efficient Dense Scene Flow from Sparse or Dense Stereo Data. In *European Conference on Computer Vision*, 2008. 2

[20] L. Xu, J. Jia, and Y. Matsushita. Motion Detail Preserving Optical Flow Estimation. In *Computer Vision and Pattern Recognition*, 2010. 2, 5

[21] A. Zaharescu, E. Boyer, K. Varanasi, and R. P. Horaud. Surface Feature Detection and Description with Applications to Mesh Matching. In *Computer Vision and Pattern Recognition*, 2009. 3

[22] Y. Zhang and C. Kambhamettu. On 3D Scene Flow and Structure Estimation. In *Computer Vision and Pattern Recognition*, 2001. 2