Estimation de flot optique multiframe par apprentissage profond

Pierre GODET, Aurélien PLYER, Alexandre BOULCH, Guy LE BESNERAIS

ONERA - Département Traitement de l'Information et Systèmes
8 chemin de la Hunière, 91123, Palaiseau, France
pierre.godet@onera.fr, aurelien.plyer@onera.fr
alexandre.boulch@onera.fr, guy.le_besnerais@onera.fr

Résumé – Ce papier concerne l'estimation du flot optique dans un contexte "multiframe", c'est-à-dire lorsque l'on dispose de séries de plus de deux images temporellement proches, par exemple issues d'une caméra à haute cadence. Nos travaux s'inscrivent dans la tendance récente à appliquer l'apprentissage profond à ces questions d'estimation. Nous présentons un estimateur, FlowNetStack, qui consiste globalement à mettre en entrée du réseau FlowNet une série de N>2 images, et que nous entraînons sur une base de vidéos simulées à haute cadence conçue pour l'occasion. Sur des données haute cadence, FlowNetStack permet d'obtenir des résultats meilleurs que des algorithmes de flot optique à l'état de l'art (PWC-Net) tout en ayant une structure beaucoup plus simple.

Abstract – We address multiframe optical flow estimation for video sequences yielded by high framerate cameras. We build on recent works exploiting deep convolutional neural networks. We present FlowNetStack, a simple structure which essentially consists in feeding FlowNet with sequences of N>2 frames. It is trained on a dedicated multiframe simulated video base. On high framerate sequences, FlowNetStack provides better performance than state-of-the-art algorithms (eg. PWC-Net), while having a much simpler structure.

1 Introduction

De nombreuses tâches de vision artificielle comme la super-résolution ou la navigation autonome requièrent l'estimation du champ des mouvements apparents, appelé flot optique. En supposant une intensité constante le long des trajectoires, de nombreuses méthodes ont été proposées depuis les travaux séminaux de Horn-Schunck [1] et Lucas-Kanade [2], [3] offrant un aperçu récent des progrès du domaine.

Contrairement à de nombreuses approches récentes qui visent une robustesse aux grands déplacements inter-image, nous proposons d'utiliser une caméra rapide pour limiter l'amplitude des déplacements inter-image et un algorithme multiframe pour profiter de la cohérence temporelle des séquences obtenues. Le cadre multiframe est relativement peu développé dans la littérature sur le flot optique. On peut citer cependant quelques travaux proposant d'utiliser une forme de régularisation temporelle des trajectoires : [4] utilise un modèle polynomial, alors que [5] se fonde sur une décomposition PCA. Un jeu de séquences haute cadence a été récemment proposé par [6] pour faciliter l'évaluation de méthodes de flot optique biframe.

Nous nous plaçons par ailleurs dans la tendance récente qui consiste à apprendre la tâche d'estimation du flot optique de manière supervisée en entraînant un réseau de neurones convolutif (*CNN*) sur de nombreux exemples d'apprentissage, notamment simulés. La référence [7] a, la première, proposé d'apprendre le flot optique à partir de deux images en entrée de *CNN*, définissant deux structures dites *FlowNetSimple* et *FlowNetCorr* (cf. Fig. 1) entraînées sur un jeu d'images simulées *FlyingChairs*. L'architecture *FlowNetSimple* est fortement inspirée de l'architecture U-net [8] : structure encodeur-décodeur

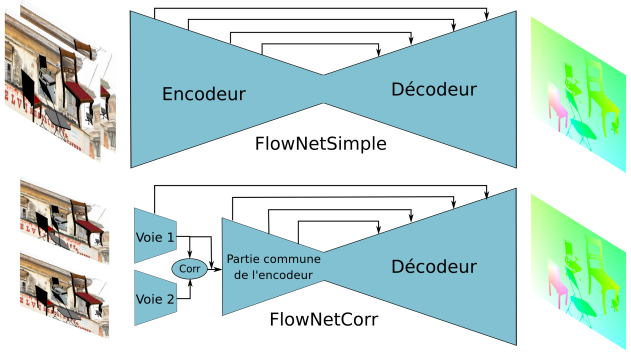

FIGURE 1 – Estimation de flot optique par apprentissage profond : architectures *FlowNetSimple* et *FlowNetCorr* [7].

avec des connections directes à chaque niveau d'échelle. De nouveaux *CNN* entraînés sur des bases de données simulées plus complexes comme *FlyingThings* [11], ont été proposés ensuite, notamment *FlowNet2* [12] et *PWC-Net* [13], ce dernier étant à l'état de l'art actuel sur plusieurs comparatifs en ligne. Ces méthodes sont rapides, estiment des flots optiques bien segmentés et sont plus robustes que les méthodes classiques dans certains cas singuliers comme le manque de texture.

Quelques stratégies multiframe commencent à voir le jour au sein de ces méthodes : [14] et [15] utilisent des flots estimés à des instants antérieurs, après les avoir recalés dans la géométrie courante, pour améliorer l'estimation du flot courant. Ces travaux restent limités à l'utilisation de N=3 images, même si [15] mentionne quelques résultats jusqu'à 5 images. Nos travaux consistent à exploiter la cohérence temporelle d'une série d'images plus longue pour améliorer les résultats d'estimation

de flot optique et, à terme, estimer des trajectoires.

2 Approche proposée

2.1 L'architecture *FlowNetStack*

Nous proposons une architecture simple basée sur *FlowNetSimple* [7] mais prenant une séquence de plus de 2 images en entrée. Dans *FlowNetSimple* les deux images d'entrée sont "empilées" de sorte que pour des images à 3 canaux (RVB) le tenseur d'entrée contienne 6 canaux. Nous proposons d'empiler non plus 2 mais N images pour donner une séquence plus longue en entrée du réseau. Afin de diminuer la taille du tenseur d'entrée les images sont passées en niveaux de gris de sorte qu'un canal corresponde à un instant. Le tenseur d'entrée a donc, dans ce cas, N canaux. Pour cela le nombre de canaux des noyaux de convolution de la première couche du réseau doit être adapté. D'autre part, la sortie du réseau ne sera plus un flot simple à 2 coordonnées mais une liste de N flots, le nombre de canaux de sortie de l'architecture doit donc également être modifié, de sorte à avoir un tenseur de sortie à $2N$ canaux. Cette nouvelle architecture est appelée *FlowNetStack*.

La dimension temporelle n'apparaît qu'à la première et à la dernière couche de *FlowNetStack*. Dans toutes les couches intermédiaires l'information temporelle est agrégée dans les différents descripteurs. L'ordre des images utilisé à l'entraînement impose l'ordre à utiliser lors de l'inférence : mélanger les images d'une séquence de test met l'algorithme en échec.

Il est possible d'utiliser des images RVB en considérant un tenseur d'entrée à $3N$ canaux. Cela serait couteux et n'apporterait qu'un gain faible en performances.

Pour entraîner cette architecture, nous utilisons le jeu d'apprentissage *ChairsMultiframe* (cf. section suivante) et la méthode d'apprentissage décrite par [7]. La fonction de coût multiframe est obtenue en sommant sur les instants la fonction biframe utilisée par [7].

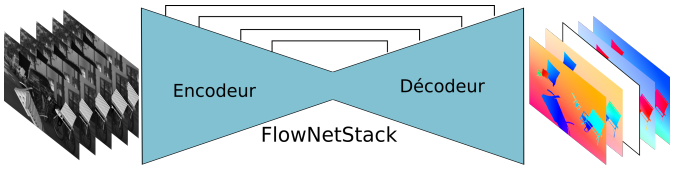

FIGURE 2 – Architecture *FlowNetStack*.

2.2 Données d'apprentissage

Le jeu de données d'entraînement *FlyingChairs* [7] ne contient que des paires d'images. *FlyingThings* [11] contient des séquences mais les mouvements mis en jeu sont de grande amplitude et compliqués pour une première phase d'entraînement (en effet, [12] montre qu'il est plus efficace de commencer par apprendre sur des données simples).

Nous avons donc généré un nouveau jeu de données synthétique *ChairsMultiframe*, inspiré par *FlyingChairs* mais en générant des séquences haute cadence à la place des paires d'images. Les images sont générées à partir de 9744 images de chaises [16] (696 modèles différents et 14 vues pour chaque) et 525 images de fonds issues de Flickr, Cityscape [17] et xview. Les fonds sont animés par des transformations homographiques définies par les déplacements rectilignes uniformes indépendants des 4 coins. Le mouvement des chaises est composé de rotations, mises à l'échelle et translations dont les dépendances temporelles sont des fonctions B-splines.

Pour cette première version 5000 séquences 320 x 240 ont été générées, avec 33 images par séquence, l'image centrale étant l'image de référence choisie pour la vérité terrain du flot optique multiframe. Par la suite une version plus conséquente pourra être générée avec les mêmes codes de génération.

3 Résultats

Le flot multiframe estimé par *FlowNetStack* peut être évalué à tous les instants en un point donné, ce qui correspond à la trajectoire de ce point pendant la séquence ; ou bien dans tout le champ spatial en un instant donné, ce qui permet la comparaison avec des méthodes biframes.

3.1 Entraînement des réseaux

Les données utilisées pour l'entraînement sont les séquences de *ChairsMultiframe* (décrites plus haut). Ces séquences comportent chacune 33 images, l'instant pris pour référence pour la vérité terrain du flot optique est l'image centrale. Pour nos expériences nous choisissons d'utiliser 7 images par séquence, en conservant l'image de référence au centre, et en sous-échantillonnant temporellement d'un facteur 5 afin d'obtenir des amplitudes de déplacements d'ordre pixellique entre deux instants consécutifs.

Trois architectures de réseaux sont entraînées sur ces mêmes données : *FlowNetStack* ainsi que les méthodes biframes *FlowNetSimple* [7] et *PWC-Net* [13] (état de l'art actuel pour l'estimation de flot biframe et monoculaire sur les challenges MPI Sintel [9] et Kitti2015 [10]). Les méthodes biframes sont entraînées sur les paires d'images au centre de chaque séquence (l'image de référence et celle qui la suit).

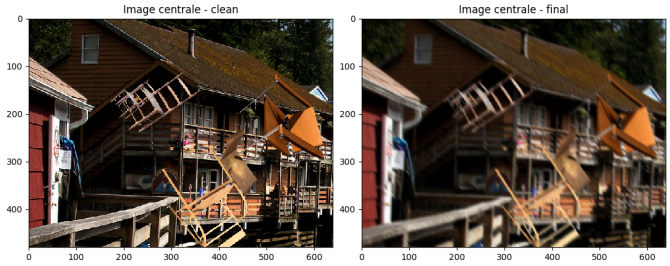

FIGURE 3 – Images de la première séquence de *ChairsMultiframeVal* en version "Clean" à gauche et "Finale" à droite.

3.2 Données pour l'évaluation

Pour l'évaluation, 12 séquences supplémentaires sont générées avec le code de *ChairsMultiframe* en prenant des images de fonds et de chaises différentes, et en générant de nouvelles trajectoires avec les mêmes distributions statistiques pour les tirages aléatoires. Les séquences comportent également 33 images, avec l'instant de référence au centre. Pour l'évaluation, deux pas de sous-échantillonnage seront considérés : 5 comme pour les séquences d'apprentissage et 2 pour s'intéresser aux mouvements sous-pixelliques, sans réentraîner les modèles. Enfin, un flou gaussien asymétrique et un bruit additif gaussien sont ajoutés à ces séquences. Ce jeu de 12 séquences est appelé *ChairsMultiframeValFinal*, cf. Fig.3, droite. Les mêmes séquences ont été générées sans flou et sans bruit, version *ChairsMultiframeValClean*, cf. Fig.3, gauche.

3.3 Comparaison avec les méthodes biframes

On s'intéresse à l'instant central du flot multiframe estimé par *FlowNetStack*, correspondant au flot optique entre l'image de référence et celle qui la suit, que l'on compare aux flots des méthodes biframes. Deux mesures d'erreur sont utilisées : l'erreur *endpoint* qui correspond à l'erreur sur la norme du flot, et l'erreur relative qui est une erreur *endpoint* normalisée localement par la norme du flot vérité terrain. Ces deux mesures d'erreur sont moyennées sur tout le champ spatial et sur les 12 séquences de validation. Deux sous-échantillonnages des mêmes séquences sont considérés afin d'avoir des déplacements pixelliques (1 image sur 5) ou sous-pixelliques (1 sur 2).

Sur la séquence pixellique (1 image sur 5), voir tableau 1, en comparant les résultats pour *FlowNetSimple* et *FlowNetStack* on observe des performances nettement améliorées avec la méthode multiframe. Ceci montre que pour un même choix d'architecture, l'ajout d'images supplémentaires améliore l'estimation. PWC-Net semble être aussi performant que *FlowNetStack* lorsqu'on s'intéresse à l'erreur *endpoint* uniquement, cependant en erreur relative l'écart se creuse en faveur de *FlowNetStack*. Cela suggère que PWC-Net rencontre plus de difficultés pour estimer précisément les mouvements de faible amplitude.

Pour comparer plus spécifiquement les estimations des mouvements de faible amplitude on considère la séquence mieux échantillonnée temporellement, voir tableau 2. Cette fois-ci *FlowNetStack* l'emporte plus largement, même contre PWC-Net. L'utilisation d'une méthode multiframe permet donc une amélioration notable pour l'estimation des mouvements de faible amplitude. [12] notait déjà que les petits mouvements posaient problème aux réseaux proposés par [7].

La figure 4 permet de comparer visuellement les flots obtenus par les différentes méthodes sur une séquence du jeu de test. Le déplacement en chaque pixel est représenté en couleur en codant la norme par la saturation et la direction par la teinte. Ces résultats sont donnés pour la séquence aux mouvements sous-pixelliques. *FlowNetStack* (en bas à droite) présente une allure spatiale du mouvement plus juste. En particulier, les structures spatiales fines ou peu texturées des deux chaises les plus à droite sont mieux estimées avec cette méthode multiframe. Le mouvement du fond, très bruité dans les estimations biframes, est nettement mieux estimé par *FlowNetStack*. Ce mouvement étant uniforme dans le temps, on pouvait s'attendre à cette amélioration en augmentant le nombre d'images considérées.

TABLEAU 1 – Erreurs moyennes sur *ChairsMultiframeVal* avec un sous-échantillonnage temporel d'un facteur 5 :

ChairsMultiframeValClean	endpoint [px]	relative [%]
FlowNetS (biframe)	0.83	52.0
PWC-Net (biframe)	0.68	40.3
FlowNetStack (multiframe)	0.67	27.4
ChairsMultiframeValFinal	endpoint [px]	relative [%]
FlowNetS (biframe)	1.04	75.5
PWC-Net (biframe)	0.82	58.4
FlowNetStack (multiframe)	0.78	46.4

TABLEAU 2 – Erreurs moyennes sur *ChairsMultiframeVal* avec un sous-échantillonnage temporel d'un facteur 2 :

ChairsMultiframeValClean	*endpoint* [px]	relative [%]
FlowNetS (biframe)	0.40	59.9
PWC-Net (biframe)	0.34	47.7
FlowNetStack (multiframe)	0.24	24.9
ChairsMultiframeValFinal	*endpoint* [px]	relative [%]
FlowNetS (biframe)	0.51	86.4
PWC-Net (biframe)	0.44	77.0
FlowNetStack (multiframe)	0.30	48.6

3.4 Évaluation des trajectoires

L'estimation multiframe permet également d'observer les trajectoires sur tout l'horizon temporel considéré. La figure 5 compare la trajectoire estimée par *FlowNetStack* à la trajectoire vérité terrain, pour un point du fond et un point d'une chaise. Pour le fond comme pour la chaise l'allure de la trajectoire est bonne. On peut cependant noter un certain biais sur la direction du mouvement du point du fond, et une amplitude légèrement sous-estimée pour le mouvement de la chaise.

4 Conclusion

Nous avons constitué un nouveau jeu de données simulé, composé de nombreuses séquences haute cadence, permettant l'entraînement de *CNN* pour l'estimation de flot optique multiframe. Nous proposons une extension simple d'une architecture biframe existante pour considérer des séquences plutôt que des paires d'images et estimer la trajectoire de chaque point. Cette méthode semble d'une part tirer profit de la redondance temporelle pour améliorer l'estimation à chaque instant, notamment dans le cas de mouvements sous-pixelliques ; et permet d'autre part une première estimation satisfaisante des tra-

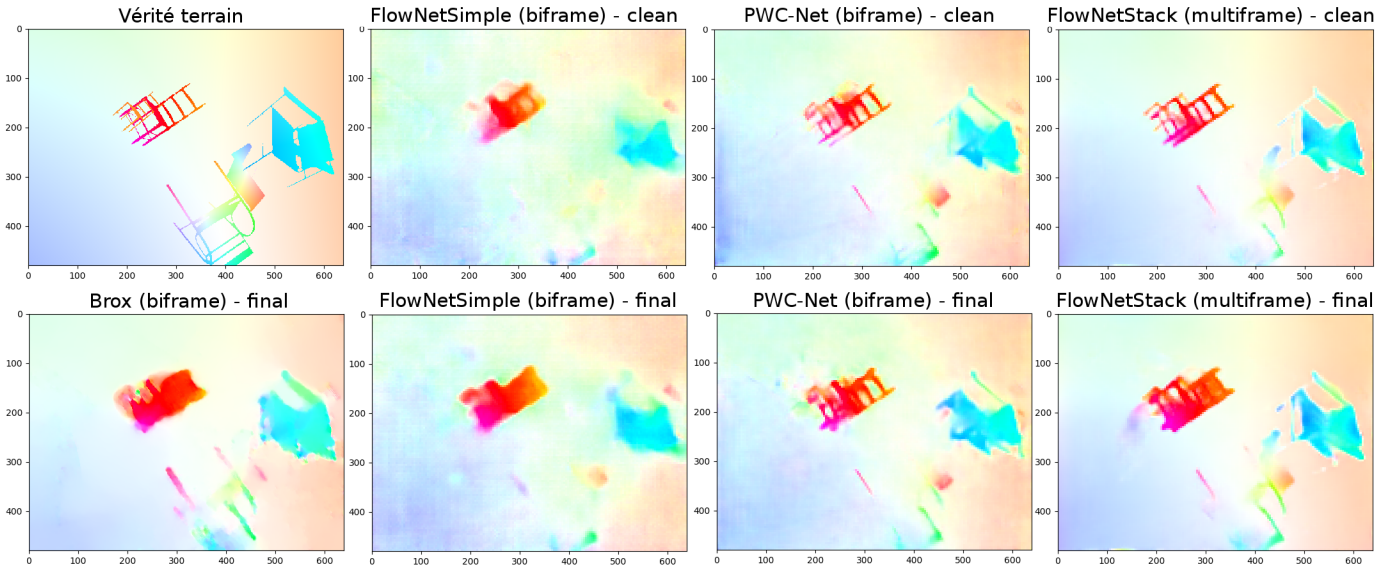

FIGURE 4 – Flots optiques estimés sur la version "Clean" (en haut) et "Finale" (en bas) de la première séquence de *ChairsMultiframeVal*, avec sous-échantillonnage d'un facteur 2. En bas à gauche, approche variationnelle de Brox et al. [18].

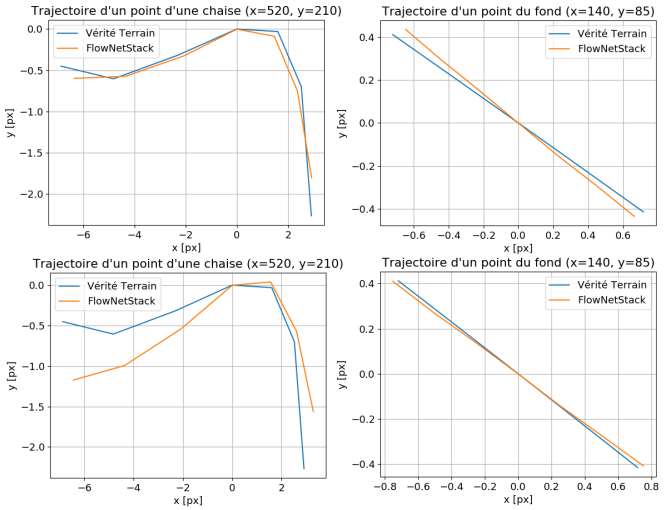

FIGURE 5 – Trajectoires estimées par *FlowNetStack*, sur la première séquence de *ChairsMultiframeVal*, avec sous-échantillonnage d'un facteur 2, en version "Clean" en haut et "Finale" en bas.

jectoires. Les travaux futurs concernent l'application à des données réelles, on s'attend à un transfert de l'apprentissage sur données simulées comme dans [7].

Remerciements : Ces travaux sont cofinancés par la Direction Générale de l'Armement.

Références

[1] B. K. Horn et B. G. Schunck. *Determining optical flow*. Artificial intelligence, 1981.

[2] B. D. Lucas et T. Kanade. *An iterative image registration technique with an application to stereo vision*. 1981.

[3] D. Sun, S. Roth et M. J. Black. *A quantitative analysis of current practices in optical flow estimation and the principles behind them*. IJCV, 2014.

[4] R. Yegavian, B. Leclaire, F. Champagnat, C.Illoul et G. Losfeld. *Lucas–Kanade fluid trajectories for time-resolved PIV*. Measurement science and Technology, 2016.

[5] R. Garg, L. Pizarro, D. Rueckert et L. Agapito. *Dense multi-frame optic flow for non-rigid objects using subspace constraints*. ACCV, 2010.

[6] J. Janai, F. Güney, J. Wulff, M. J. Black et A. Geiger. *Slow Flow : Exploiting High-Speed Cameras for Accurate and Diverse Optical Flow Reference Data*. CVPR, 2017.

[7] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, et T. Brox. *Flownet : Learning optical flow with convolutional networks*. ICCV, 2015.

[8] O. Ronneberger, P. Fischer et T. Brox. *U-net : Convolutional networks for biomedical image segmentation*. MICCAI, 2015.

[9] D. J. Butler, J. Wulff, G. B. Stanley et M. J. Black. *A naturalistic open source movie for optical flow evaluation*. ECCV, 2012.

[10] M. Menze and A. Geiger. *Object Scene Flow for Autonomous Vehicles*. CVPR, 2015.

[11] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy et T. Brox. *A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation*. CVPR, 2016.

[12] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, T. Brox. E. Ilg et collab. *Flownet 2.0 : Evolution of optical flow estimation with deep networks*. CVPR, 2017.

[13] D. Sun, X. Yang, M. Y. Liu et J. Kautz. *Pwc-net : Cnns for optical flow using pyramid, warping, and cost volume*. CVPR, 2018.

[14] M. Neoral, J. Šochman et J. Matas. *Continual Occlusions and Optical Flow Estimation*. arXiv :1811.01602, 2018.

[15] Z. Ren, O. Gallo, D. Sun, M. H. Yang, E. B. Sudderth et J. Kautz. *A Fusion Approach for Multi-Frame Optical Flow Estimation*. arXiv :1810.10066, 2018.

[16] A. X. Chang et collab. *Shapenet : An information-rich 3d model repository*. arXiv :1512.03012, 2015.

[17] M. Cordts et collab. *The cityscapes dataset for semantic urban scene understanding*. CVPR (proceedings), 2016.

[18] T. Brox, A. Bruhn, N. Papenberg et J. Weickert. *High accuracy optical flow estimation based on a theory for warping*. ECCV, 2004.