

Estimation du mouvement et de la structure à l'échelle absolue à partir d'un réseau multi-caméras non synchronisées

R. MHIRI¹ P. VASSEUR¹ S. MOUSSET¹ R. BOUTTEAU² A. BENSRAHAIR¹

¹ LITIS - Université de Rouen - INSA de Rouen

² IRSEEM - ESIGELEC

Résumé

Nous proposons un système de vision, basé sur un réseau de caméras non-synchronisées permettant d'estimer le mouvement d'un véhicule et la structure de l'environnement 3D à l'échelle absolue. L'algorithme proposé nécessite au moins trois images prises par au moins deux caméras. Les poses relatives des caméras sont estimées par les méthodes classiques d'estimation du mouvement. Les déplacements absolus sont calculés en supposant que les trajectoires entre deux vues consécutives d'une même caméra sont rectilignes et que les caméras sont calibrées hors ligne. Une étape d'optimisation par ajustement de faisceaux est réalisée pour affiner l'estimation des facteurs d'échelle et les positions des points 3D. Notre méthode est évaluée à partir de données réelles issues de la base de données KITTI.

Mots Clef

Estimation du mouvement, réseau multi-caméra non synchronisées, facteur d'échelle.

Abstract

We propose an unsynchronized camera network able to estimate the motion of a vehicle and the 3D structure with accurate absolute scale. The proposed algorithm requires at least three frames acquired from two cameras. The relative camera poses are estimated with classical Structure-from-Motion and the absolute scales between views are computed by assuming straight trajectories between consecutive views of one camera and extrinsic parameters from two cameras. We propose a final optimisation step to refine only the scale and the 3D points. Our method is evaluated in real conditions on the KITTI dataset. We show quantitative evaluation through comparisons against GPS/INS ground truth.

Keywords

motion estimation, asynchronous multi camera network, scale factor.

1 Introduction

Les longues séquences d'images représentent un défi important pour l'estimation de mouvement d'un véhicule. En effet, les petites erreurs issues du processus d'estimation sont accumulées itérativement lors de la séquence, ce qui provoque une dérive dans les trajectoires estimées. Ce problème peut être traité par un ajustement de faisceaux (AF). Si il est appliqué pour une séquence entière, cela correspond à un ajustement de faisceaux global, et si il est appliqué localement pour certaines images, ce sera un ajustement de faisceaux fenêtré ou local. Par ailleurs, l'estimation de la structure et du mouvement avec l'estimation du facteur d'échelle absolu est aussi un défi important. Pour estimer le facteur d'échelle absolu, les systèmes monoculaires nécessitent une étape d'initialisation ou une connaissance a priori sur le système et/ou la structure. Les systèmes multi-caméras peuvent être une bonne alternative pour l'estimation de l'échelle tout en permettant de couvrir toute la zone autour d'un véhicule [1]. Toutefois, si la géométrie épipolaire est nécessaire pour certains algorithmes comme dans [2], la synchronisation des caméras devient indispensable, or cette solution technique nécessite un investissement important. Un réseau de caméras asynchrone présente plusieurs avantages. Tout d'abord, un tel système peut être développé avec des dispositifs à faible coût pour être commercialisés à grande échelle, ce qui est souvent suggéré pour les applications réelles [3]. Dans ce cas, l'acquisition ne dépend plus de la caméra la plus lente ou du dispositif de synchronisation lui-même. Les images peuvent être acquises en continu à partir de chaque caméra séparément. La contrainte de bande passante qui apparaît dans le cas de caméras synchronisées peut être également contournée, enfin, le réseau peut être facilement modifié.

Dans notre travail précédent, nous avons développé une nouvelle méthode appelé la méthode des triangles, qui n'impose aucune contrainte ni sur la synchronisation des caméras ni sur leurs positions relatives [4]. Cette méthode est basée sur l'hypothèse de linéarité de la trajectoire entre deux images consécutives d'une même caméra. L'algo-

ritme proposé considère trois images, dont deux issues de la même caméra et la troisième d'une caméra voisine. Avec des algorithmes classiques, nous avons estimé les poses relatives entre ces images. L'ensemble des poses et de la structure ne peut être estimé qu'à un facteur d'échelle près. La connaissance de la distance entre les deux caméras (estimée par un étalonnage hors ligne) combinée avec une hypothèse de mouvement rectiligne du système, nous a permis d'estimer les facteurs d'échelle absolus. Bien que la méthode des triangles soit une méthode robuste, certaines erreurs se produisent dans les virages. Ces erreurs sont engendrées par l'hypothèse de linéarité et réduisent la précision de l'estimation de l'échelle absolue. L'imprécision de l'estimation des facteurs d'échelle entraîne de grandes erreurs sur l'estimation du mouvement et de la structure. Pour améliorer la précision de l'échelle absolue, nous proposons une optimisation des facteurs d'échelle. Ainsi le système multi-caméras asynchrone devient plus complet et plus précis. Les principales contributions présentées dans ce papier se résument dans la minimisation des erreurs imposées par l'hypothèse de mouvement rectiligne de la méthode des triangles. Pour mettre l'accent sur l'optimisation de l'échelle absolue, nous supposons que les matrices de rotation et les vecteurs de translations estimés initialement par l'algorithme de 5-points [5] sont fixes et nous n'optimisons que les facteurs d'échelle et la structure 3D.

Le reste du document est organisé comme suit : dans la section 2, nous présentons une brève discussion sur les travaux antérieurs liés à l'estimation de mouvement et au processus d'optimisation. Ensuite, nous donnons une description détaillée de la méthode des triangles et l'ajustement de faisceaux appliqué à notre réseau de caméras dans la section 3. Avant de conclure notre travail dans la dernière section, des expérimentations et des résultats réels sont présentés dans la section 4.

2 Travaux antérieurs

Les algorithmes d'estimation de mouvement ont été largement étudiés [6] [3] [7]. L'estimation de pose à partir des mises en correspondances 2D-2D peut être obtenue avec l'algorithme « 5-points » [5]. Une estimation itérative à partir des points d'intérêt provoque toujours des erreurs qui seront accumulées au fil du temps et conduisent à des dérives que même les algorithmes robustes ne peuvent pas éviter. Pour remédier à ce problème, un processus d'optimisation tels que l'AF est généralement appliqué pour optimiser l'estimation du mouvement et de la structure 3D. Pour les applications d'odométrie visuelle, une fermeture de boucle suivie par un ajustement de faisceau global est généralement appliquée [8]. Cette technique est très efficace mais difficile à mettre en place dans des applications réelles. Cependant, il est possible d'utiliser un ajustement de faisceaux local sur une fenêtre temporaire pour limiter les calculs [9][10][11].

Les systèmes de caméras asynchrones ont été rarement étudiés pour les problèmes d'estimation de mouvement.

Dans [12], un système stéréo de caméra non synchronisées a été présenté afin d'estimer la structure 3D avec des techniques basées sur le SLAM. Dans un environnement intérieur, un robot effectue la reconstruction 3D à l'aide de deux caméras non synchronisées. Cette méthode utilise trois images : deux provenant de la caméra à gauche au premier et au troisième instant et une image à partir de la caméra droite au deuxième instant. Les primitives des trois images sont interpolées afin de créer l'image gauche manquante au deuxième instant en supposant que la position de ces primitives change linéairement entre les images. En utilisant les paramètres de l'odomètre du robot, cette méthode permet d'avoir une structure 3D précise. Cependant, Cette approche traite un cas asynchrone mais pas les méthodes d'estimation de mouvement d'un système de caméras asynchrones.

Lorsque l'échelle métrique absolue est souhaitée, le processus d'odométrie visuelle doit intégrer une connaissance 3D particulière. Pour les systèmes monoculaires, une connaissance a priori de la structure 3D est nécessaire comme initialisation afin de maintenir l'échelle absolue dans l'estimation de la séquence. Dans [7], Fraundorfer et al. présentent une paramétrisation d'AF contraint pour l'estimation de l'échelle. Pour calculer les poses relatives, les auteurs utilisent l'algorithme de 1-point présenté dans [13]. Cet algorithme est basé sur le principe de direction Ackermann pour les mouvements circulaires. Pour calculer l'échelle, Fraundorfer et al. proposent une paramétrisation d'un AF global pour résoudre tous les facteurs d'échelle à la fois. Au lieu d'optimiser tous les paramètres du mouvement et de la structure comme l'algorithme classique d'un AF, ils optimisent seulement les facteurs d'échelle relatifs. La principale différence entre [7] et notre travail réside dans l'utilisation d'un système de caméra monoculaire et nous avons développé un système multi-caméras asynchrone. De plus, leur algorithme est basé sur l'hypothèse de mouvement circulaire et le notre est basé sur une hypothèse de mouvement rectiligne.

3 Estimation du mouvement et de la structure

La méthode proposée est divisée en deux parties principales : l'estimation du mouvement et de la structure à l'échelle métrique absolue et l'optimisation des facteurs d'échelle absolue et de la structure par AF.

3.1 Méthode des triangles

Dans notre travail précédent [4], l'approche repose sur deux hypothèses : deux images consécutives d'une même caméra suivent une trajectoire rectiligne et les caméras voisines partagent un champ de vue commun. Notre méthode peut être généralisée à N caméras et nécessite au moins deux caméras. Dans le reste de l'article, nous présentons l'approche pour deux caméras C_i et C_j . Les poses relatives des caméras sont calculées via les techniques de l'estimation du mouvement et de la structure (Structure-from-

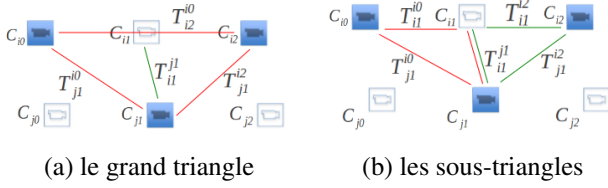


FIGURE 1 – Méthode des triangles pour des caméras non synchronisées, une caméra bleu est une caméra qui acquiert une image : C_{i0} , C_{j1} , et C_{i2} . C_{i1} est la position virtuelle de la caméra C_i à l’instant 1.

Motion, SFM). Les facteurs d’échelle absolue sont calculés en utilisant le calibrage extrinsèque des caméras et l’hypothèse de linéarité.

Les trois poses relatives représentées par les matrices de rotation R et les vecteurs unitaires de translation t sont calculées entre les trois images du triangle (figure 1) en utilisant l’algorithme « 5-points ». La figure 1 (a) montre la forme du triangle entre la caméra C_i à l’instant 0 et à l’instant 2, et la caméra C_j à l’instant 1. Les lignes rouges représentent les transformations entre les positions des caméras obtenues par SFM et la ligne verte représente la transformation issue du processus d’étalonnage extrinsèque (basé sur une mire de calibrage). T_{i2}^{i0} est la transformation du repère de la caméra C_{i2} dans le repère de la caméra C_{i0} , de même pour T_{j1}^{i0} et T_{i2}^{j1} . La transformation statique T_{i1}^{j1} représente le couplage statique des deux caméras à un même instant due à la rigidité du système.

Nous intégrons la transformation statique T_{i1}^{j1} afin de calculer les facteurs d’échelle absolue. La pose de la caméra virtuelle C_i à l’instant 1, C_{i1} , est intercalée dans le grand triangle. Cette pose donne naissance à deux sous-triangles (figure 1(b)). Dans les trois triangles obtenus, nous pouvons écrire les équations suivantes :

$$\begin{cases} T_{i1}^{i0} = T_{j1}^{i0} T_{i1}^{j1} \Rightarrow \begin{bmatrix} R_{i1}^{i0} & \lambda_1 t_{i1}^{i0} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_{j1}^{i0} & \alpha t_{j1}^{i0} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_{i1}^{j1} & t_{i1}^{j1} \\ 0 & 1 \end{bmatrix} \\ T_{i2}^{i0} = T_{j1}^{i0} T_{i2}^{j1} \Rightarrow \begin{bmatrix} R_{i2}^{i0} & \lambda_2 t_{i2}^{i0} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_{j1}^{i0} & \beta t_{j1}^{i0} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_{i2}^{j1} & t_{i2}^{j1} \\ 0 & 1 \end{bmatrix} \\ T_{i2}^{i0} = T_{j1}^{i0} T_{i2}^{j1} \Rightarrow \begin{bmatrix} R_{i2}^{i0} & (\lambda_1 + \lambda_2) t_{i2}^{i0} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_{j1}^{i0} & \alpha t_{j1}^{i0} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_{i2}^{j1} & \beta t_{i2}^{j1} \\ 0 & 1 \end{bmatrix} \end{cases} \quad (1)$$

Le développement de ce système d’équations et le découplage des termes en rotation et des termes en translation permettent d’écrire l’équation matricielle suivante :

$$\begin{bmatrix} t_{i1}^{i0} & 0 & -t_{j1}^{i0} & 0 \\ 0 & t_{i2}^{i0} & 0 & -t_{j1}^{i0} \\ t_{i2}^{i0} & t_{i2}^{i0} & -t_{j1}^{i0} & -R_{j1}^{i0} t_{i2}^{j1} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} R_{j1}^{i0} t_{i1}^{j1} \\ R_{j1}^{i0} t_{i2}^{j1} \\ 0 \end{bmatrix} \quad (2)$$

Les facteurs d’échelle absolue λ_1 , λ_2 , α , et β peuvent être calculés en considérant l’équation 2 comme un modèle linéaire de moindres carrés. Le système peut être écrit sous

la forme $A.X = B$ avec X le vecteur des facteurs d’échelle. La translation de la caméra C_{i0} dans le repère de la caméra C_{i1} et la translation de C_{i2} dans le repère de C_{i1} sont obtenus grâce à l’hypothèse de linéarité. La méthode des triangles est décrites en détails dans [4].

3.2 Ajustement de faisceaux

L’AF a un rôle important dans les applications de vision par ordinateur basées sur la reconstruction 3D et le SFM. Il permet d’affiner les paramètres du mouvement et la structure 3D décrivant l’environnement [14]. C’est une étape d’optimisation à la fois des positions 3D et des paramètres des caméras. L’AF consiste à minimiser les erreurs de re-projection entre les points détectés et les points projetés dans l’image à partir de la structure 3D. Dans notre approche, nous proposons de réaliser un AF local, c’est à dire sur un nombre de vues limité. Pour mettre en avant l’optimisation des facteurs l’échelle estimés par la méthode des triangles, nous supposons que les matrices de rotation et les vecteurs de translation estimés initialement sont fixes et nous n’optimisons que les facteurs d’échelle et la structure 3D. Nous avons choisi d’appliquer l’AF sur une fenêtre glissante composée de deux triangles consécutifs (cinq images). Nous calculons les points 3D à l’échelle à partir des points mis en correspondance et des poses des caméras exprimées dans le repère de la première caméra de chaque fenêtre.

Formulation du problème. Dans le modèle sténopé, la fonction de projection d’un point 3D X de la scène dans le plan de l’image (un point 2D x) peut être écrite en utilisant une transformation perspective : $x = K[R|s * t]X$ avec K la matrice des paramètres intrinsèques, $[R|s * t]$ la matrice des paramètres extrinsèques et s le facteur d’échelle.

L’AF minimise les erreurs de re-projection entre les points 2D détectés et les points 2D projetés à partir des points 3D. La minimisation de l’erreur de re-projection est résolue en utilisant un algorithme non linéaire des moindres carrés tel que Levenberg-Marquardt [15]. Cet algorithme permet d’obtenir des approximations successives d’un vecteur de paramètres P . Selon l’algorithme de Levenberg-Marquardt (algorithme 1), Δ est obtenu en résolvant l’équation normale augmentée à chaque itération. Les erreurs de re-projection sont calculées et évaluées à la fois pour P_i et P_{i+1} .

Calcul de la jacobienne. Dans notre méthode, la matrice jacobienne J est calculée en dérivant la fonction de projection par le facteur d’échelle s et les points 3D X seulement. En effet, la différenciation symbolique de la fonction de projection F est effectuée par rapport à s et aux coordonnées du point 3D X . En différenciant la fonction de projection par rapport à X , nous obtenons la matrice JX (matrice 2x3) comme dans l’équation 3. Lorsque nous dérivons la fonction de projection par rapport aux facteurs d’échelle, nous obtenons la matrice JS (matrice 2x3).

$$JX = \left[\frac{\partial F}{\partial X} \right] \text{ et } JS = \left[\frac{\partial F}{\partial s} \right]. \quad (3)$$

Algorithm 1 Pseudo-code de l'algorithme de Levenberg-Marquardt

```

i ← 0
λ ← 0.001
calcul de ||e(P0)||
while i < MAX_ITERATIONS et ||e(Pi)|| > seuil do
  résolution de l'équation normale augmentée :
  (JTJ + λI)Δ = -JTe
  évaluation du vecteur de paramètres Pi+1 = Pi + Δi
  if ||e(Pi+1)|| ≥ ||e(Pi)|| then
    λ ← 10λ
  end if
  if ||e(Pi+1)|| ≤ ||e(Pi)|| then
    λ ← 10λ, Pi = Pi+1
  end if
  i ← i + 1
end while

```

Pour chaque point 3D, nous calculons les matrices JX et JS pour toutes les caméras de la fenêtre glissante. La jacobienne J aura une structure éparse (figure 2). Si l'on considère n caméras et m les points 3D, la jacobienne sera une matrice $(2 * n * m) \times (1 * n + 3 * m)$.

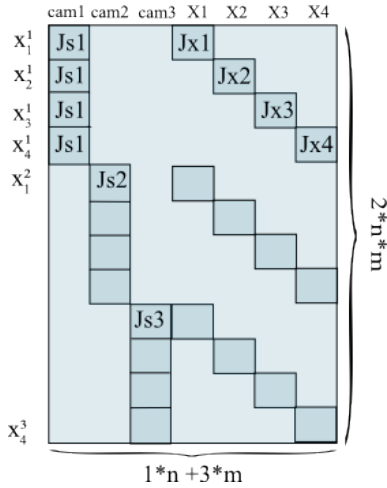


FIGURE 2 – Structure d'une matrice jacobienne pour une fenêtre composée de trois caméras et 4 points 3D. Le bleu foncé désigne les paramètres des caméras JS et les paramètres des points 3D JX et le bleu clair représente les éléments nuls.

4 Résultats

Dans cette section, nous présentons les résultats de notre méthode appliquée sur une séquence d'images réelles de la base de données KITTI [16] [17]. Les capteurs utilisés dans notre évaluation sont des caméras stéréo parfaitement synchronisées et un système de navigation inertielle GPS/IMU. Ce dernier capteur permet d'avoir une vérité terrain. Pour avoir l'aspect asynchrone, nous utilisons une seule image sur la paire stéréo à chaque instant.

Pour chaque série de trois images (trois instants), les points d'intérêt sont extraits en utilisant le détecteur SURF [18] et décrits avec le descripteur Freak [19]. Les appariements entre deux images permettent d'estimer une matrice essentielle grâce à l'algorithme « 5-points ». Les rotations et les translations relatives sont obtenues à partir des matrices essentielles estimées. Les points d'intérêt retenus comme « inliers » sont ensuite triangulés afin d'estimer les positions des points 3D. Le système d'équation 2 est résolu pour calculer les facteurs d'échelle absolue λ_1 , λ_2 , α et β . Pour deux triangles consécutifs (cinq images), nous calculons les points 3D et les poses des caméras à l'échelle et nous les exprimons dans le repère de la première caméra de la fenêtre glissante. Ensuite, nous appliquons l'algorithme de Levenberg Marquard. Nos résultats sont présentés pour quelques images de la séquence 0 de la base de données KITTI.

4.1 Résultats d'un ajustement de faisceaux « optimal »

Pour l'évaluation quantitative de notre algorithme, nous appliquons l'AF en utilisant les poses de la vérité terrain (VT). Pour une fenêtre glissante de deux triangles, nous utilisons les matrices de rotation, les vecteurs unitaires de translations et les facteurs d'échelle obtenus par le GPS comme entrée de l'AF. Ensuite, nous calculons les points 3D à l'échelle en utilisant les données de la vérité terrain puis nous appliquons l'étape de l'AF. Ce test, que nous appelons un ajustement de faisceaux « optimal », nous servira comme référence pour évaluer les facteurs d'échelle estimés après l'application de l'AF aux données estimées. Pour évaluer les facteurs d'échelle de l'AF optimale, nous calculons le rapport du facteur d'échelle évalué par le facteur d'échelle de la VT, avant et après l'AF comme dans les équations 4 et 5.

Nous obtenons pratiquement les mêmes données de départ avec de petites erreurs que nous jugeons acceptables. Puis, nous ajoutons un bruit gaussien aux valeurs des facteurs d'échelle issus des poses du GPS, la trajectoire obtenue en accumulant les poses après l'AF est presque que celle de départ (la trajectoire obtenue par le GPS). Les résultats sont présentés dans le tableau 1 avec l'échelle 1 le facteur d'échelle de la pose de la deuxième caméra de la fenêtre dans le repère de la première caméra de la fenêtre, l'échelle 2 le facteur d'échelle de la pose de la troisième caméra dans le repère de la première, de même pour l'échelle 3 et 4.

$$\text{ratios avant} = \frac{\text{facteur d'échelle évalué avant AF}}{\text{facteur d'échelle de la VT}} \quad (4)$$

$$\text{ratios après} = \frac{\text{facteur d'échelle évalué après AF}}{\text{facteur d'échelle après l'AF optimale}} \quad (5)$$

Si les ratios calculés sont proches de 1 alors nous jugeons que notre algorithme donne des résultats très précis. Les

TABLE 1 – Ratios avant et après l’AF pour la VT bruitée par un bruit gaussien ($\sigma = 0.01$)

	échelle 1	échelle 2	échelle 3	échelle 4
avant AF	1.0014	0.9993	0.9996	0.9994
après AF	1.0003	0.999803	01.0001	1.003

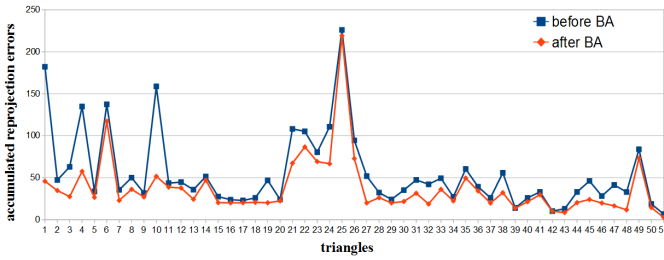


FIGURE 3 – Les erreurs de reprojection pour 52 triangles avant et après un AF appliqué sur les données estimées par la méthode des triangles

erreurs sont très faibles et dues à plusieurs raisons : l’imprécision des détecteurs et des descripteurs des points d’intérêt, les erreurs de mise en correspondance et les erreurs de calcul de la triangulation et de la reprojection des points 3D dans les images. L’AF permet de minimiser les erreurs de reprojection pour améliorer les facteurs d’échelle et les points 3D. Lorsque les paramètres du mouvement sont parfaits (VT), quelques erreurs restent encore présentes sur la structure, c’est pourquoi nous comparons les résultats obtenus par la méthode des triangles par rapport à un AF « optimal » afin de quantifier correctement les facteurs d’échelle obtenus.

4.2 Résultats de la méthode des triangles

Dans cette section, nous présentons les résultats de l’AF sur l’estimation issue de la méthode des triangles. La figure 4 montre la distribution des erreurs de reprojection accumulées avant et après l’AF pour une séquence de 200 images. Nous calculons les ratios de la même façon que les équations 4 et 5. Les résultats sont résumés dans le tableau 2. Les trajectoires sont présentées dans la figure 6. La figure montre que la trajectoire optimisée par la méthode proposée est plus proche de la VT que la trajectoire obtenue par la méthode des triangles avant l’AF.

5 Conclusion

Dans cet article, nous avons présenté une méthode d’estimation de mouvement complète que nous avons appelé la méthode des triangles en utilisant une configuration multicaméras non synchronisées. L’approche présentée dans notre travail précédent [4] suppose que la trajectoire entre deux images consécutives issues d’une même caméra est rectiligne et nécessite une connaissance de la calibration



FIGURE 4 – Exemple d’erreurs de reprojection des points 3D après AF, chaque couleur fait référence aux erreurs de reprojection dans une caméra de la fenêtre glissante (5 caméras)

TABLE 2 – Ratios avant et après un AF appliqué sur l’estimation initiale par la méthode des triangles

	échelle 1	échelle 2	échelle 3	échelle 4
Avant AF	0.9516	0.944478	0.9555	0.9500
Après AF	1.037	1.002	1.0286	1.0609

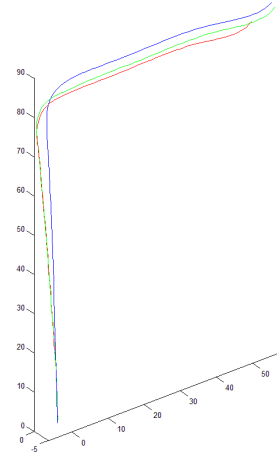


FIGURE 5 – Trajectoires obtenues à partir de 200 images : l’estimation en rouge, la VT en bleu, l’AF en vert

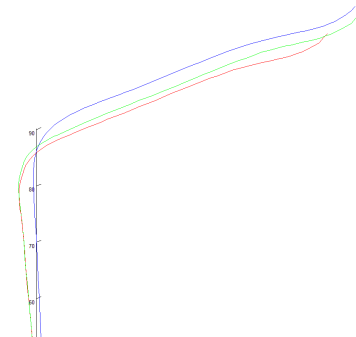


FIGURE 6 – zoom sur les trajectoires

géométrique effectuée hors-ligne. L'hypothèse de linéarité provoque de petites erreurs en particulier pour les virages. Nous avons appliqué un AF sur les facteur d'échelle absolue et sur la structure 3D pour affiner l'estimation initiale. Les résultats de l'approche présentée améliore la précision et la robustesse de l'estimation des facteurs d'échelle absolue. Cela permet de reconstruire le déplacement d'un véhicule à l'échelle réelle.

Annexe

Ce travail est sponsorisé par le projet ANR Blanc DrAAcar N ANR-11-IS03-0003 et la Région Haute-Normandie.

Références

- [1] P. Lébraly, *Etalonnage de caméras a champs disjoints et reconstruction 3D : Application À un robot mobile*. PhD thesis, Université Blaise Pascal - Clermont-Ferrand II, 2006.
- [2] T.-W. Hui, "Structure from motion directly from a sequence of binocular images without explicit correspondence establishment," in *Image Processing (ICIP), 2013 20th IEEE International Conference on*, pp. 3607 – 3611, IEEE, 2013.
- [3] P. Furgale, U. Schwesinger, M. Ruffi, W. Denderdarz, H. Grimmitt, P. Muhlfellner, S. Wonneberger, J. Timpner, S. Rottmann, B. Li, *et al.*, "Toward automated driving in cities using close-to-market sensors : An overview of the v-charge project," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pp. 809–816, IEEE, 2013.
- [4] R. Mhiri, P. Vasseur, S. Mousset, R. Bouteau, and A. Bensrhair, "Visual odometry with unsynchronized multi-cameras setup for intelligent vehicle application," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pp. 1339–1344, IEEE, 2014.
- [5] D. Nistér, "An efficient solution to the five-point relative pose problem," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 6, pp. 756–770, 2004.
- [6] G. H. Lee, F. Faundorfer, and M. Pollefeys, "Motion estimation for self-driving cars with a generalized camera," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 2746–2753, IEEE, 2013.
- [7] F. Fraundorfer, D. Scaramuzza, and M. Pollefeys, "A constricted bundle adjustment parameterization for relative scale estimation in visual odometry," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 1899–1904, IEEE, 2010.
- [8] M. Lhuillier, "Automatic scene structure and camera motion using a catadioptric system," *Computer Vision and Image Understanding*, vol. 109, no. 2, pp. 186–203, 2008.
- [9] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd, "Real time localization and 3d reconstruction," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1, pp. 363–370, IEEE, 2006.
- [10] C. Engels, H. Stewénus, and D. Nistér, "Bundle adjustment rules," in *In Photogrammetric Computer Vision*, 2006.
- [11] E. O.-B. Alfredo Ramirez and M. Trivedi, "Panoramic stitching for driver assistance and applications to motion saliency-based risk analysis," in *Intelligent Transportation Systems (ITSC 2013)*, IEEE, 2013.
- [12] M. Svedman, L. Goncalves, N. Karlsson, M. Munich, and P. Pirjanian, "Structure from stereo vision using unsynchronized cameras for simultaneous localization and mapping," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pp. 3069–3074, IEEE, 2005.
- [13] D. Scaramuzza, F. Fraundorfer, and R. Siegwart, "Real-time monocular visual odometry for on-road vehicles with 1-point ransac," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pp. 4293–4299, IEEE, 2009.
- [14] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment a modern synthesis," in *Vision algorithms : theory and practice*, pp. 298–372, Springer, 2000.
- [15] K. Levenberg, "A method for the solution of certain problems in least squares," *Quarterly of applied mathematics*, vol. 2, pp. 164–168, 1944.
- [16] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving ? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
- [17] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics : The kitti dataset," *International Journal of Robotics Research (IJRR)*, 2013.
- [18] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf : Speeded up robust features," in *Computer Vision–ECCV 2006*, pp. 404–417, Springer, 2006.
- [19] A. Alahi, R. Ortiz, and P. Vanderghenst, "Freak : Fast retina keypoint," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 510–517, Ieee, 2012.