Descripteur sémantique géométrique pour le matching de points dans les applications d'odométrie visuelle

Malick KANDJI[1,2] Sébastien AMBELLOUIS[1] Cyril MEURIE[1] Cédrick LELIONNAIS[2] Charles TATKEU[1]

[1]Univ Gustave Eiffel, COSYS-LEOST, F-59650 Villeneuve d'Ascq, France

[2]SNCF Voyageurs, Centre d'Ingénierie du Matériel ; France

Résumé – Pour des applications d'odométrie visuelle (VO) et de SLAM, nous proposons un nouveau descripteur sémantique géométrique de points d'intérêt. Ce descripteur combine des informations sur la géométrie locale et une représentation quantitative des classes sémantiques avoisinantes. Il est conçu pour être invariant au point de vue, ce qui améliore la robustesse face aux changements d'orientation et de perspective. En complément, un préfiltrage d'image est appliqué au besoin dans les régions à faible texture afin d'améliorer la détection de points. Nos expérimentations menées sur la base de données publique automobile KITTI-360 démontrent que notre descripteur améliore les performances de méthodes de référence issues de la littérature telles que ORB-SLAM2 en configuration stéréo et VO, *c.-à-d.,* sans fermeture de boucle et optimisation globale. Par ailleurs, de par sa nature, ce descripteur sémantico-géométrique s'intègre aisément à n'importe quel type d'algorithme de VO ou de SLAM.

Abstract – For visual odometry (VO) and SLAM applications, we propose a novel semantic-geometric keypoint descriptor. This descriptor combines local geometric information with a quantitative representation of surrounding semantic classes. It is designed to be viewpoint-invariant, thereby enhancing robustness to changes in orientation and perspective. Additionally, an image pre-filtering step is applied, when necessary, in low-texture regions to improve keypoint detection. Our experiments, conducted on the public automotive dataset KITTI-360, demonstrate that the proposed descriptor improves the performance of state-of-the-art methods such as ORB-SLAM2 in both stereo and VO configurations, *i.e.*, without loop closure or global optimization. Furthermore, due to its inherent structure, the semantic-geometric descriptor can be seamlessly integrated into any VO or SLAM algorithm.

1 Introduction

Dans le domaine de la robotique et de la navigation autonome, la localisation précise d'un véhicule est une étape essentielle pour garantir des déplacements sûrs et efficaces. Pour y parvenir, plusieurs méthodes ont été proposées, notamment les algorithmes de SLAM (*Simultaneous Localization and Mapping*) et d'odométrie visuelle (*Visual Odometry*, VO).

L'odométrie visuelle (OV) permet d'estimer, à partir d'une séquence d'images, la pose (position et orientation) d'un agent mobile à chaque instant par rapport à un référentiel initial. Elle se compose de deux étapes. La partie *front-end* permet d'extraire des caractéristiques de l'image, de les mettre en correspondance et d'estimer le déplacement. La partie *back-end* optimise les poses estimées dans une fenêtre glissante afin de produire une carte locale.

Le SLAM étend les capacités de la VO en y ajoutant une optimisation globale de l'ensemble des poses estimées ainsi qu'une étape de reconnaissance de lieux (*loop closure*) permettant de corriger les erreurs accumulées au fil du temps, notamment lorsque l'agent revisite un endroit déjà parcouru.

Cependant, ces méthodes sont sensibles à certaines sources d'erreurs, telles que :
(1) les images faiblement texturées, mal éclairées ou présentant des motifs répétitifs, qui dégradent l'étape d'extraction des caractéristiques,
(2) les mauvaises mises en correspondance (stéréoscopiques ou temporelles), dues par exemple à la présence d'objets dynamiques autour du véhicule.

Pour surmonter ces difficultés, diverses approches ont été proposées. Certaines reposent sur des méthodes classiques de traitement d'images et d'optimisation, tandis que d'autres font appel à des architectures neuronales capables d'estimer certains éléments de la chaîne VO, tels que la profondeur, le flot optique, la pose, les points d'intérêt et leurs descripteurs, ou encore la segmentation sémantique. Cette dernière reste relativement peu explorée, bien qu'il ait été démontré que l'information sémantique peut améliorer les mises en correspondance et, par conséquent, la qualité de l'estimation odométrique.

Dans cet article, nous proposons une approche visant à renforcer la robustesse des mises en correspondance de points d'intérêt en combinant des informations géométriques locales avec le contexte sémantique environnant (ex. : bâtiments, véhicules, piétons). Cette combinaison est formalisée sous la forme d'un nouveau descripteur sémantico-géométrique. Une étape de préfiltrage est également introduite pour améliorer la détection des points d'intérêt dans les zones à faible texture.

L'article est structuré en trois parties : nous commençons par une revue de la littérature sur la VO et le SLAM, suivie de la description de notre approche, incluant le nouveau descripteur et le préfiltrage, avant de présenter une évaluation sur la base d'images KITTI-360 [4].

2 Travaux existants

De manière générale, pour un système visuel pur soumis à des bruits de mesure, le problème de VO est posé sous une forme probabiliste en maximisant le maximum a posteriori (MAP) équivalent à maximiser la vraisemblance, *c.-à-d.,*

$$\theta^* = argmax_\theta P(Z/\theta) \quad (1)$$

où $\theta \in \mathbb{R}^6$ désigne la pose à estimer, et Z représente les observations.

Dans ce cadre, deux grandes familles d'approches coexistent : les méthodes indirectes et les méthodes directes.

Les méthodes indirectes reposent sur l'extraction de points d'intérêt (FAST, SIFT, ORB, SuperPoint) ou de lignes puis à la génération d'un vecteur de caractéristiques pour chacun d'entre eux. Une fois ces informations extraites dans les images, elles sont mises en correspondance sur le plan stéréoscopique (dans le cas d'un système à deux caméras). Cette étape permet de produire un nuage de points 3D utilisé pour assurer une tâche de suivi temporel à partir de laquelle la pose est estimée. ORB-SLAM2 [5] est l'une des méthodes de référence exploitant l'extracteur ORB.

Plus récemment, des extracteurs basés sur des architectures neuronales profondes ont été proposés : SuperPoint [1] offre une meilleure invariance qu'ORB mais nécessite l'utilisation d'un GPU pour assurer une extraction rapide.

Les méthodes directes utilisent directement l'intensité lumineuse des pixels des images puis effectuent un alignement photométrique pour estimer la pose. Cet alignement peut être dense si tous les pixels sont utilisés ou semi-dense en utilisant uniquement les pixels à fort gradient.

Dans la suite, nous nous intéressons aux méthodes indirectes pour lesquelles la détection et la mise en correspondance des points est cruciale pour assurer un bon suivi temporel, une bonne estimation de la pose et un fonctionnement sans interruption *c.-à-d.,* sans perte du suivi temporel.

Dans cette optique, *Lianos et al.* proposent VSO [3] pour réduire les erreurs liées à l'étape de suivi des points en intégrant la contrainte sémantique dans l'optimisation de la pose de la carte locale. Plus récemment, *Ilter et al.* [2] ont intégré un descripteur sémantique dans la fonction de mise en correspondance pour réduire les erreurs d'appariement. Ce dernier est modélisé par un histogramme binaire représenté par la présence ou non de certaines classes dans le voisinage des points d'intérêt. Cependant, ce descripteur réduit à un histogramme n'intègre aucune information spatiale et géométrique liée à la sémantique autour du point d'intérêt.

Dans ce papier, nous proposons de compléter ce descripteur sémantique en ajoutant la contrainte géométrique afin notamment de le rendre plus robuste à la rotation et d'améliorer la qualité de la mise en correspondance.

Par ailleurs, pour garantir la détection d'un nombre minimal de points d'intérêt et ainsi éviter les interruptions de l'estimateur, une étape de préfiltrage supplémentaire est ajoutée en combinant plusieurs filtres dans les zones à faible contraste ou à faible texture où aucun point d'intérêt n'est détecté.

Notre descripteur peut être intégré à n'importe quel processus de VO et de SLAM. Dans notre cas, nous l'avons intégré à ORB-SLAM2 [5], en mode odométrie en configuration stéréo, *i.e.,* sans fermeture de boucle et optimisation globale (*Bundle Adjustment*), comme montré sur la figure 1.

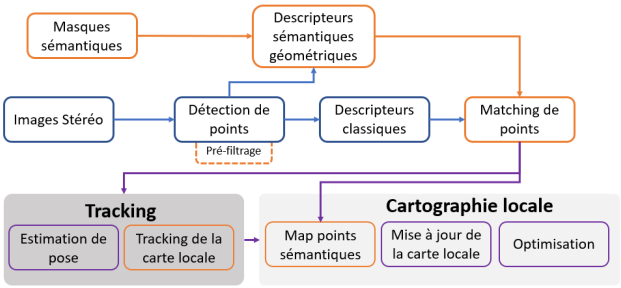

FIGURE 1 : **Processus de la méthode de VO**. Les étapes en orange correspondent aux modules ajoutés : segmentation sémantique, description sémantique quantitative et géométrique, préfiltrage, mise en correspondance intégrant les contraintes sémantico-géométriques, ainsi que la mise à jour des points de la carte locale avec ces informations sémantiques.

3 Méthode proposée

Nous proposons deux points d'amélioration de l'algorithme ORB-SLAM2 dans sa version VO.

La première a lieu durant la phase d'extraction de caractéristiques. Lorsque les images contiennent des régions avec peu de texture ou à faible luminosité, il est parfois impossible d'y détecter des points d'intérêt. Dans ce cas, nous proposons d'y appliquer un préfiltrage afin d'améliorer localement cette étape d'extraction. Le préfiltrage est constitué de la moyenne pondérée de plusieurs images filtrées avec un filtre CLAHE utilisant un seuil adaptatif, un filtre de Sobel, un filtre Laplacien et une égalisation d'histogramme. Cette fusion permet d'améliorer le contraste et d'augmenter la luminosité. Cette étape est d'autant plus importante car un enjeu crucial est d'uniformiser la distribution des points d'intérêt dans l'image pour éviter les zones surchargées ou vides.

La seconde amélioration consiste à définir un nouveau descripteur sémantique plus robuste aux transformations par rotation et utilisable lors des étapes de mise en correspondance stéréoscopique et de suivi temporel.

Soit S_i le masque de segmentation sémantique associé à une image I_i. Ce masque représente le résultat de la segmentation sémantique sur cette image. Pour un ensemble de classes sémantiques N_C, chaque label c est associé à un masque binaire $S_i^{(c)}$. Une illustration d'un masque superposé à son image originelle est présentée figure 2.

Soit P_i l'ensemble des points détectés sur une image I_i :

$$P_i = \left\{(p_{i,j}, s_{i,j}, \theta_{i,j}) \mid p_{i,j} \in \mathbb{R}^2, \quad s_{i,j} \in \mathbb{R}, \quad \theta_{i,j} \in [0, 2\pi[\right\}$$

avec $j = \{0, ..., n\}$; $p_{i,j}$, $s_{i,j}$ et $\theta_{i,j}$ représentant respectivement le j^{me} point détecté sur une image i, son label sémantique et l'angle d'orientation associé.

Nous ne considérons que les catégories sémantiques présentes dans l'histogramme binaire de description sémantique tel que proposé par [2] dans le masque $S_i^{(c)}$ du voisinage de chaque point $p_{i,j}$.

Pour chaque point $p_{i,j}$ détecté dans une image i, on définit un voisinage circulaire de rayon r. Nous calculons le barycentre, noté $G_{i,j}^{(c)}$, de chaque classe sémantique présente dans le voisinage de la manière suivante :

$$G_{i,j}^{(c)}(x,y) = \left(\frac{\sum\limits_{x,y} x \cdot S_{i,j}^{(c)}(x,y)}{\sum\limits_{x,y} S_{i,j}^{(c)}(x,y)}, \frac{\sum\limits_{x,y} y \cdot S_{i,j}^{(c)}(x,y)}{\sum\limits_{x,y} S_{i,j}^{(c)}(x,y)} \right) \quad (2)$$

Ensuite, le point obtenu est centré et normalisé afin de borner toutes les valeurs de distance dans l'intervalle $[0, 1]$:

$$\begin{cases} x' = (x - c_x)/2r \\ y' = (-y + c_y)/2r \end{cases} \quad (3)$$

avec (c_x, c_y) représentant les coordonnées du point central (*i.e.,* $p_{i,j}$).

Pour être robuste au changement de point de vue, une rotation de θ est appliquée avec la matrice de rotation $R(\theta)$.

Un exemple de point est illustré sur la figure 2.

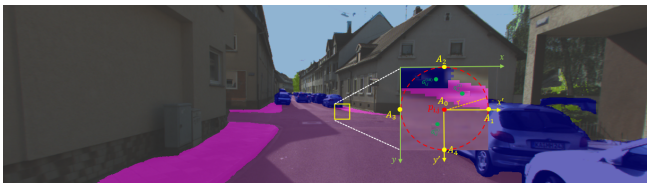

FIGURE 2 : Point d'intérêt avec les barycentres des masques en vert et les points fixes en jaune. Image : superposition d'une frame et de son masque de segmentation sémantique associé.

Une fois les barycentres calculés et transformés, leurs normes L_2 sont calculées par rapport à des points fixes préalablement définis qui sont au voisinage du point d'intérêt. Dans un souci d'optimisation du temps de calcul, 5 points fixes sont choisis. Le premier est le point d'intérêt au centre $A_0(0;0)$. Les 4 autres sont sur le cercle délimitant le voisinage du point : $A_1(0.5;0)$, $A_2(0;0.5)$, $A_3(-0.5;0)$ et $A_4(0;-0.5)$.

Ainsi, pour chaque point d'intérêt, en plus du descripteur de point classique, nous rajoutons le descripteur sémantique géométrique qui est une matrice de taille $N_C \times N_F$, N_F étant le nombre de points fixes. Chaque ligne représente une classe sémantique et chaque colonne la norme L_2 entre le barycentre et le point fixe correspondant

Si une classe sémantique est absente pour un point d'intérêt, les distances associées sont mises à 0.

Pour faire le matching des points, la norme L_2 est calculée entre les descripteurs sémantiques géométriques des deux points. Cette dernière est bornée entre 0 et $N_C \cdot N_F$, étant donné que les valeurs de chaque élément des matrices sont entre 0 et 1. Les distances du descripteur classique sont fusionnées avec celle de notre descripteur et celle introduite par [2]. Pour cela, nous faisons un changement d'échelle et rajoutons un poids α_2 pour pondérer l'impact de la distance d_{sg} :

$$d = (1 - \alpha_1 - \alpha_2)d_p + \alpha_1 \frac{|d_p|}{N_C} d_s + \alpha_2 \frac{|d_p|}{N_C \cdot N_F} d_{sg} \quad (4)$$

avec d_p la distance du point d'intérêt et $|d_p|$ sa valeur maximale, d_s la distance sémantique proposée par [2], N_C et N_F *resp.* le nombre de classes sémantiques et le nombre de points fixes.

Dans le cas de ORB-SLAM2, la distance de Hamming est utilisée pour les features ORB et sa valeur maximale est de 256 tandis que pour SuperPoint la norme L_2 est utilisée, avec une valeur maximale de $\sqrt{2}$ dans la mesure où les valeurs du descripteur sont normalisées.

4 Résultats expérimentaux

Dans cette section, nous analysons l'impact du nouveau descripteur sémantique géométrique intégré à la partie VO d'ORB-SLAM2 et évalué sur la dataset automobile KITTI-360.

Trois configurations sont considérées : l'utilisation exclusive d'ORB (ORB-VO), de SuperPoint (SP-VO), ou la combinaison des deux (ORB-SP-VO). Toutes les configurations sont en mode stéréo, avec un total de 3000 points d'intérêt détectés pour ORB et SP. Dans le cas d'ORB-SP-VO, 1500 points de chaque type sont utilisés. Les points ORB sont extraits sur 8 niveaux d'échelle, tandis que ceux de SuperPoint le sont sur 4. Les coefficients α_1 et α_2, déterminés empiriquement, sont fixés à 0.1 et restent constants sur l'ensemble de toutes les séquences.

KITTI-360 est une dataset automobile de $73.7km$ de long, constituée de 9 longues séquences avec des données multi-capteurs provenant notamment d'un système stéréoscopique. Nous avons testé les algorithmes sur toutes les 9 séquences mais n'avons pas réussi à traiter l'intégralité de la séquence 02.

En raison du caractère non déterministe introduit par le multi-threading et certaines étapes d'optimisation, chaque exécution de la méthode sans apprentissage (ORB-VO) a été répétée trois fois, puis la médiane des résultats a été conservée. En revanche, SP-VO et ORB-SP-VO n'ont été exécutés qu'une fois, en raison de leur coût computationnel élevé sur les longues séquences de KITTI-360. L'inférence a été réalisée sur des paires stéréo à l'aide de deux GPU fonctionnant en parallèle.

Le modèle SuperPoint utilisé n'a pas été ré-entraîné sur KITTI-360 ; les poids pré-entraînés sur la dataset HPatches sont directement exploités. Pour la segmentation sémantique, nous avons recours à PSPNet [6], fine-tuné sur KITTI-360 à partir du modèle initialement entraîné sur Cityscapes, avec 19 classes.

L'évaluation des performances des différentes approches de VO repose sur la cohérence locale des trajectoires estimées, mesurée à l'aide de la métrique $RPE\ 1m(\%)$, qui quantifie la dérive relative moyenne tous les 1 m parcourus à l'aide du $RMSE$ intra-séquence.

TABLE 1 : Résultats sur KITTI-360 avec les features ORB. ORB-VO (OVO) : algorithme par défaut avec les features ORB. +DS (Descripteur Semantique de [2]). +P (Préfiltrage). +Nous2 : OVO+DS+DSG (Descripteur Semantique Géométrique). +Nous3 : OVO+DS+DSG+P. Les meilleurs résultats par séquence sont mis en **gras** et les seconds meilleurs scores soulignés. wru : sans les usagers de la route (voitures, piétons, etc.) qu'ils soient dynamiques ou non.

Seq	ORB-VO				
	OVO	+P	+DS	+Nous2	+Nous3
00	2.113	2.011	2.053	1.982	**1.942**
03	2.362	2.421	2.346	2.282	**2.240**
04	2.311	2.209	2.304	2.278	**2.147**
05	11.62	2.558	4.64	11.20	**2.17**
05 wru	3.062	2.966	3.083	3.034	**2.888**
06	2.353	2.225	2.350	2.303	**2.164**
07	**34.59**	34.42	35.22	35.21	35.11
07 wru	2.697	2.594	2.620	**2.584**	2.668
09	14.09	14.37	14.12	14.26	**14.00**
10	6.240	**6.142**	6.351	6.352	6.249

Les tableaux 1 et 2 présentent les résultats obtenus sur les séquences de KITTI-360. De manière générale, notre approche améliore la précision des différents algorithmes considérés individuellement : ORB-VO, SP-VO et ORB-SP-VO affichent des performances supérieures dans la majorité des cas lors-

TABLE 2 : Résultats avec les features SP et ORB+SP.

Seq	SP-VO			ORB-SP-VO	
	SVO	+DS	+Nous2	OSVO	+Nous2
00	2.598	2.932	2.822	2.129	**2.099**
03	2.391	2.265	**2.247**	2.391	2.320
04	3.462	5.194	5.718	2.426	**2.376**
05	19.81	2.810	**2.240**	47.16	26.39
06	2.850	2.499	**2.318**	2.424	2.430
07	33.91	**30.19**	31.31	–	–
09	14.15	15.32	14.20	**13.96**	14.38
10	6.408	6.412	6.307	**6.298**	6.388

qu'ils sont associés à notre descripteur sémantico-géométrique et notre préfiltrage.

Dans le cas d'ORB-VO, l'ajout du descripteur et du préfiltrage (configuration OVO+Nous3 dans le tableau 1) conduit à une amélioration significative sur les séquences 00, 03, 04, 05, 06 et 09, avec un gain global en précision supérieur à 8%. En excluant la séquence 07, particulièrement difficile, cette amélioration dépasse 24%.

Le préfiltrage, bien qu'il soit limité aux cellules de l'image où aucun point n'a été détecté (typiquement dans des zones à faible texture), permet un rehaussement ciblé destiné à faciliter la détection de points. Cette approche améliore la couverture dans des environnements complexes tels que les routes, les tunnels, la végétation ou les surfaces peu texturées. Son efficacité se manifeste particulièrement sur la séquence 10, marquée par de nombreux passages en tunnel et une faible luminosité.

SP-VO souffre quant à lui de pertes fréquentes du suivi des points, dues notamment à des vibrations importantes ou à des changements brusques de luminosité à la sortie de tunnels (figure 3(b)). Ces difficultés sont accentuées par le fait que le modèle n'a pas été adapté au jeu de données (absence de *fine-tuning*). Malgré cela, l'intégration de notre méthode dans SP-VO conduit à une amélioration de la précision, tendance également observée avec ORB-SP-VO. À noter que dans la séquence 09, SP-VO échoue à maintenir le suivi en raison d'un mouvement trop rapide de la caméra.

Certaines dégradations de performance sont liées à la présence d'objets dynamiques (voitures, camions), en particulier dans les séquences 05, 07 et 10. La séquence 07 se révèle particulièrement problématique : la dérive observée pour tous les algorithmes est due à la présence prolongée d'un véhicule (figure 3(a)) sur lequel se concentrent la majorité des points détectés. Lorsque ces objets mobiles sont exclus (configuration *wru* dans les tableaux), les performances s'en trouvent notablement améliorées. Un traitement similaire a été appliqué à la séquence 05, en raison d'un camion roulant devant le véhicule.

Enfin, les résultats obtenus avec la combinaison directe des descripteurs ORB et SP dans ORB-SP-VO suggèrent qu'une fusion plus intelligente, exploitant les complémentarités de chaque méthode, pourrait permettre de meilleures performances.

Une piste à explorer serait le développement d'un modèle d'apprentissage capable d'estimer automatiquement des poids de fusion optimaux, adaptatifs aux conditions de l'environnement, et intégrables directement dans l'architecture ORB-SP-VO.

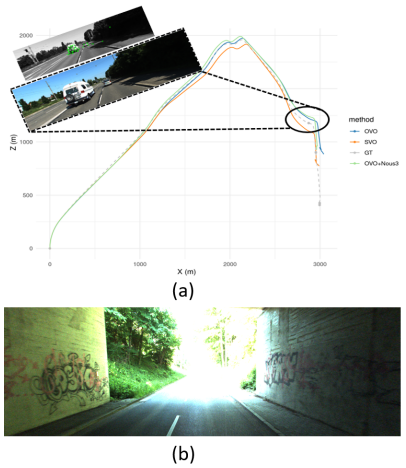

FIGURE 3 : (a) Trajectoires séquence 07. (b) Sortie de Tunnel.

5 Conclusion

Dans cet article, nous proposons une nouvelle méthode qui améliore la précision des algorithmes de VO et de SLAM en modifiant la partie *front-end*. D'abord un rehaussement de l'image est effectué avec un filtrage adaptatif puis le matching est robustifié en intégrant les contraintes sémantiques, avec l'information géométrique sémantique et quantitatif sur le voisinage du point. Cette méthode, pouvant être intégrée à n'importe quel algorithme, a été évaluée sur ORB-SLAM2 en configuration stéréo et VO (sans fermeture de boucle ni optimisation globale). Des points d'intérêt de type ORB et SuperPoint ont été utilisés pour l'extraction de caractéristiques. La combinaison des deux est prometteuse et dans nos prochains travaux, nous prévoyons de fusionner ces deux extracteurs en intégrant de l'apprentissage automatique et de l'information sémantique dans l'optimisation de pose.

Références

[1] Daniel DETONE, Tomasz MALISIEWICZ et Andrew RABINOVICH : Superpoint : Self-supervised interest point detection and description. *In IEEE CVPR workshops*, pages 224–236, 2018.

[2] Oguzhan ILTER, Iro ARMENI, Marc POLLEFEYS et Daniel BARATH : Semantically Guided Feature Matching for Visual SLAM. *In 2024 IEEE ICRA*, pages 12013–12019. IEEE, 2024.

[3] Konstantinos-Nektarios LIANOS, Johannes L SCHONBERGER, Marc POLLEFEYS et Torsten SATTLER : Vso : Visual semantic odometry. *In ECCV*, pages 234–250, 2018.

[4] Yiyi LIAO, Jun XIE et Andreas GEIGER : KITTI-360 : A Novel Dataset and Benchmarks for Urban Scene Understanding in 2D and 3D. *IEEE Transactions on PAMI*, 45(3):3292–3310, mars 2023.

[5] Raúl MUR-ARTAL et Juan D. TARDÓS : ORB-SLAM2 : An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, octobre 2017.

[6] Hengshuang ZHAO, Jianping SHI, Xiaojuan QI, Xiaogang WANG et Jiaya JIA : Pyramid scene parsing network. *In CVPR*, 2017.