

Détection entièrement automatique de points de fuite dans des scènes architecturales urbaines

■ Mahzad KALANTARI, Franck JUNG

En photogrammétrie architecturale ainsi que dans les programmes de levers urbains 3D qui prennent de plus en plus d'importance actuellement, nous recherchons des solutions permettant de faciliter la mise au point sur le terrain du tableau d'assemblage et de l'aérotriangulation de la façon la plus automatique possible. Or dans ce genre de chantiers il est clair qu'on peut tirer parti des perspectives formées par les droites parallèles de l'objet photographié et des points de fuites qui en résultent sur les images. Mais pour que l'opération soit efficace ces points de fuite doivent être retrouvés sans intervention humaine, problème qui jusqu'ici n'a pas été résolu de façon pleinement satisfaisante. Suite à nos travaux, il est présenté ici une nouvelle méthode de détection de points de fuite complètement automatique, robuste et de complexité réduite vis-à-vis des algorithmes classiques, qui tient en deux étapes : une détection de segments par un algorithme de Canny-Deriché modifié, suivie d'une caractérisation de chacun des segments par leur appartenance à un cercle spécifique de l'espace original. La bonne performance en termes de robustesse se double donc d'une réduction de complexité puisque l'ensemble du processus est réalisé en 2D sans aucune aide externe.

■ mots-clés

Patrimoine architectural, détection automatique de points de fuite, cercles K, photogrammétrie architecturale.

Les programmes de levers urbains 3D sont actuellement au centre de nombreuses recherches, avec comme objectif général de permettre une reconstitution la plus automatisée possible de la volumétrie des bâtiments, tout en plaquant ensuite sur les façades les images orthorectifiées issues d'acquisitions photographiques classiques obtenues au niveau de chaque rue. Lorsque ce problème est abordé avec les outils de la photogrammétrie, il s'avère très important de pouvoir garantir, dès la phase d'acquisition de terrain, que les images acquises permettent bien une restitution complète, et donc que le tableau d'assemblage résultant est satisfaisant. Et dans la même logique, il est recherché toute solution permettant de garantir la qualité de l'aérotriangulation et la possibilité de la réaliser sur le terrain. Parmi les différentes voies possibles, nous avons étudié comment rechercher les points de fuite dans les images. Les points de fuite sont en effet caractéristiques des images d'objets fabriqués par l'homme, tout particulièrement les bâtiments, dans lesquels la quasi totalité des lignes visibles sur des images correspondent à des éléments strictement horizontaux ou verticaux. Ces directions spécifiques peuvent donc être employées pour faciliter l'orientation des images, pour autant qu'on sache les y retrouver de façon simple. Dans la géométrie conique caractéristique de la vision humaine ou de la photographie, ces lignes parallèles se traduisent par des faisceaux de droites qui concourent sur des points de fuite. Classiquement, un bâtiment isolé simple donne lieu à 3 points de fuite, celui lié aux lignes verticales, et

un pour chaque groupe de lignes horizontales dans chaque façade visible. Le but de ce papier est la production d'un nouvel algorithme permettant la détection automatique de tous les points de fuite engendrés par les lignes visibles dans une image 2D. Nous débutons dans la partie 2 par un état de l'art de la détection des points de fuite qui montre un ordre de complexité élevé des algorithmes mis en jeu et un manque de robustesse à toutes les situations de l'architecture urbaine.

La méthode proposée repose sur deux parties. La partie 3 présente une première détection des contours par une adaptation du filtrage de Canny-Deriché qui associe à chaque segment trouvé un ensemble de paramètres qualifiant son incertitude. La partie 4 utilise cette information en entrée pour établir le classement des segments selon le point de fuite afin de localiser ensuite facilement celui-ci. Une validation visuelle des résultats obtenus est finalement présentée et discutée.

Etat de l'art

Il existe à ce jour différentes méthodes pour la détection des points de fuite. Une des plus importantes est basée sur l'emploi de la sphère de Gauss, avec une grande diversité de variantes depuis son introduction par Barnard [2] en 1983. L'avantage de cette méthode est de ramener dans un espace fini tous les points de fuite. Cette approche a été complétée en 1984 par Magee et Aggarwal [11] qui accumulent la projection

- de l'intersection des segments de l'image sur la sphère de Gauss. Cette méthode est très lourde au niveau des calculs mais considérée comme très précise. Elle procède d'un calcul qui intervient dans l'espace 3D de la sphère de Gauss.

En 1998, Lutton et al. [10], proposent une nouvelle approche qui est une adaptation de la transformée de Hough pour la détection des points de fuite. Tuytelaars [14] introduit peu après une méthode interactive basée également sur la transformée de Hough, sous le nom de "Cascade Hough Transform". D'autres méthodes n'utilisent pas d'espace fini d'accumulation, et travaillent directement sur l'image. C'est le cas de Quan et Mohr [13] ou Den Heuvel [9] qui ont introduit une méthode de détection basée sur des contraintes géométriques. Brauer et Voss [4] s'intéressent à la détection des points de fuites dans des images où le niveau de bruit est élevé. Récemment, Almansa [1] a développé une nouvelle méthode de détection de points de fuite qui ne nécessite aucune information a priori, mais utilise des modèles probabilistes complexes.

Notre démarche consiste à détecter de façon automatique les points de fuite dans l'espace image sans avoir recours à des espaces finis 3D, en se basant sur une géométrie simple 2D.

Détection des segments droits sur l'image

Les segments sont extraits en utilisant une approche classique. Le travail est réalisé sur une image de luminance.

Pour commencer, une détection de contours est réalisée par un filtre de Canny-Deriche [6]. Ensuite, une détection des maxima locaux dans la direction du gradient est effectuée. Une localisation subpixelaire de ces maxima est réalisée en utilisant une technique d'interpolation décrite dans [12]. Cette étape est suivie d'un seuillage par hystérésis à deux paramètres ($S_b = 5$, $S_h = 10$). Ces seuils sont pris suffisamment bas (à la limite de la distinction visuelle) afin de ne pas constituer des seuils critiques dans cet algorithme. Expérimentalement, aucun problème de détection n'a pu être observé en utilisant ces seuils. Ces contours sont chaînés. Une approximation polygonale des contours est réalisée. La polygonisation est réalisée en utilisant une fusion itérative basée sur le résidu maximum de la régression orthogonale. On fusionne d'abord les polygones dont la fusion fournit un résidu maximum minimal. L'algorithme de fusion est stoppé dès que le résidu maximum dépasse un seuil fixé. Dans nos applications et compte tenu de notre souhait d'utiliser les segments de droites dans des processus d'estimation de points de fuites ainsi que dans un processus d'orientation des images, nous avons choisi un seuil très bas : 0.5 pixels.

Cette étape de polygonation est suivie par une estimation des paramètres de la droite ainsi que de la matrice de variance covariance de ces paramètres. L'algorithme utilisé est décrit dans [5]. Notons qu'aucun de ces paramètres ne semble critique pour la démarche proposée. Il est même possible d'en estimer certains (seuil de polygonisation) en fonction du rapport signal sur bruit de l'image ainsi que du paramètre α du filtre de Canny-Deriche.

Classification des segments par groupes de gerbes perspectives

■ Caractérisation des segments extraits de l'image

Chaque segment peut être caractérisé de nombreuses façons différentes, par exemple par les coordonnées de ses points d'extrémité, mais il nous a semblé plus adapté de les caractériser par leur distance à l'origine du système de coordonnées, et par l'angle formé par la direction de la droite orthogonale au segment l'un des axes.

Pour des raisons pratiques, l'origine O a été choisie dans le coin supérieur gauche de l'image, l'axe des x étant horizontal et l'axe des y sur la verticale gauche, le repère ainsi formé n'étant donc pas de sens direct, mais chaque pixel recevant ainsi des coordonnées positives. L'angle θ est compté par rapport à l'axe x (voir figure 1). Nous désignons par la lettre H l'intersection entre la droite orthogonale issue de O au segment considéré et ledit segment.

Dans un premier temps, nous avons analysé visuellement l'histogramme formé par l'ensemble des θ de tous les segments. Un tel histogramme (cf. Figure 2) montre bien les différents groupes issus des gerbes perspectives correspondant aux différents points de fuite de l'image.

Néanmoins, si cet histogramme permet d'isoler sans aucune ambiguïté les lignes correspondant aux verticales (cf. Figure 2) pour autant que la photographie ait été acquise avec un axe plus ou moins horizontal, par contre les gerbes perspectives correspondant aux lignes horizontales des bâtiments se mélangent au moins en partie, sans qu'il soit possible de les séparer sur ce seul critère de θ .

Il suffit pour s'en persuader de prendre l'exemple de lignes horizontales du bâtiment localisées près de l'horizontale issue du centre optique : dans cette zone de l'image, deux gerbes perspectives correspondant à deux façades contiguës d'un même bâtiment sont nécessairement mélangées. Au-delà de cet exemple qui est assez ponctuel (zone de l'image limitée au voisinage de l'horizontale issue de l'axe optique), on trouve sans difficultés de nombreux groupes de segments parallèles appartenant à deux gerbes perspectives différentes, et qui rendent une discrimination sur le seul θ assez largement inopérante.

D'autres critères de sélection ont donc été recherchés. Nous nous sommes intéressés ici à une discrimination basée sur les

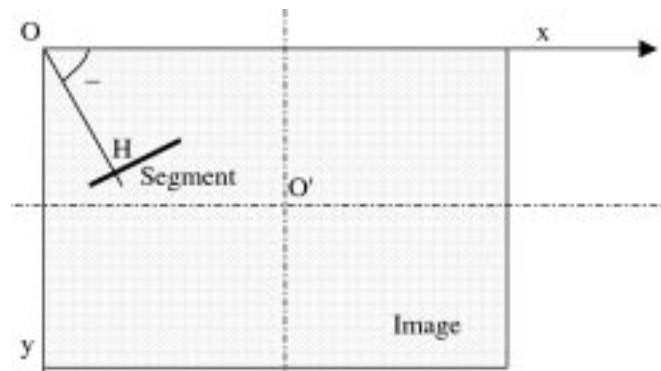


Figure 1 : Référence adoptée pour l'étude des points de fuite.

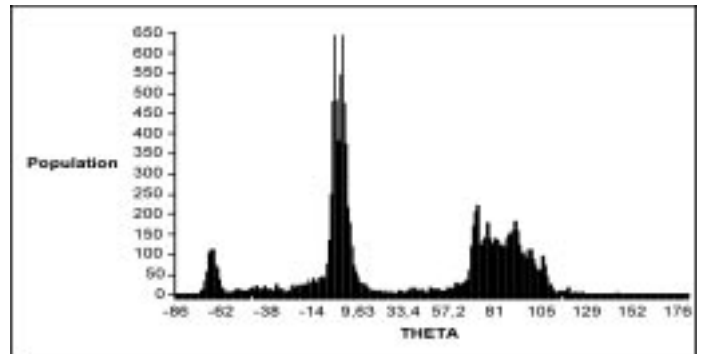


Figure 2 : Une image de segments, et son histogramme des orientations θ .

points H. En effet, pour deux segments parallèles appartenant à deux gerbes différentes, les points H sont presque toujours très différents. Nous avons donc étudié la géométrie de la figure formée par l'origine O, les points H de tous les segments, et les différents points de fuite P_i . Il est intéressant de noter que pour une gerbe perspective donnée correspondant à un point P_i , les points H de tous les segments de cette gerbe sont théoriquement disposés sur un cercle K de diamètre OP. En effet, les droites OH et HP sont orthogonales par construction, donc tous les triangles OHP sont rectangles de même hypoténuse OP. L'exploitation de cette propriété géométrique a déjà été suggérée par Bräuer & Voss [4].

Nous avons donc étudié la possibilité de regrouper l'ensemble des points H de l'image (autant que de segments), directement sous forme de cercles. En effet, il est possible d'envisager ensuite une extraction automatique de tous les cercles formés par ces points H. Une caractéristique des cercles K est d'être définis par seulement deux paramètres, puisqu'ils passent par l'origine des coordonnées. (cf. Figure 4)

■ Recherche automatique des cercles K

Le problème est maintenant de définir tous les cercles qui passent par des ensembles significatifs de points H. Un cercle est défini par 3 points, et il existe évidemment différentes façons de faire passer un cercle par un ensemble de points, soit en minimisant la distance algébrique entre le cercle et les différents points, soit en minimisant la distance géométrique [8]. Dans notre application et pour la simplicité du modèle nous avons choisi de minimiser la distance algébrique :

$$F(x) = a x^2 + b^t x + c = 0, \quad (1)$$

où $a \neq 0$ et $b \in \mathbb{R}^2$.

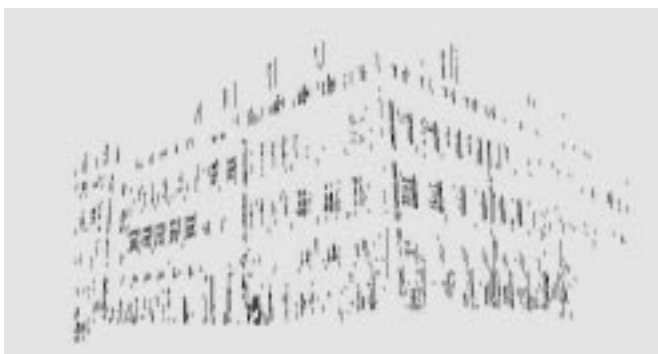


Figure 3 : Classification des segments verticaux par rapport à leurs angle θ .

Pour ajuster un cercle à un nuage de points, il faut calculer a, b et c. En insérant les coordonnées des points dans l'équation (1) nous obtenons un système d'équations tel que :

$$B u = 0, \quad (2)$$

où $u = (a ; b_1 ; b_2 ; c)$ et

$$B = \begin{bmatrix} X^2_{i1} + X^2_{i2} & X^2_{i3} & X^2_{i4} & X^2_{i5} \\ \dots & \dots & \dots & \dots \\ X^2_{in1} + X^2_{in2} & X^2_{in3} & X^2_{in4} & X^2_{in5} \end{bmatrix}. \quad (3)$$

Pour trouver la solution de l'équation homogène (2) nous imposons la contrainte suivante :

$$\|u\| = 1, \quad (4)$$

et nous cherchons donc à minimiser le système suivant :

$$\min_{\|u\|=1} \|B u\|. \quad (5)$$

Ce système sera résolu par une décomposition SVD et la solution finale sera celle de la plus petite valeur propre du vecteur propre correspondant :

$$K = (k_1, k_2) = \left(-\frac{b_1}{2a}, -\frac{b_2}{2a} \right) \quad (6)$$

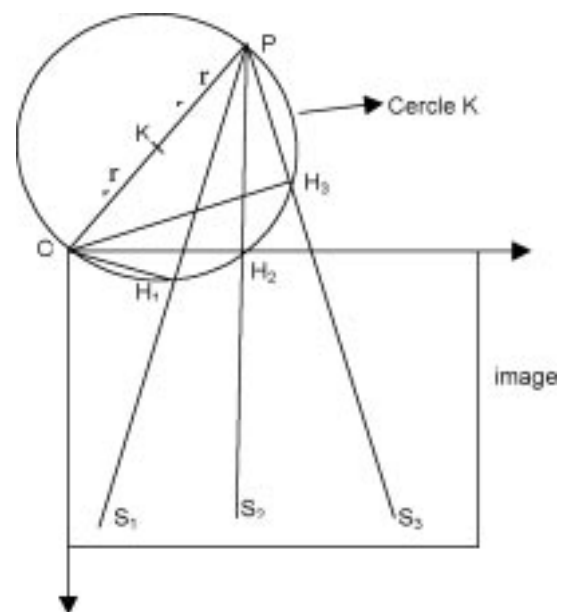


Figure 4 : Configuration des points H, du point de fuite et du cercle K sur l'image.

$$r = \sqrt{\frac{|b|^2}{4a^2} - \frac{c}{a}} \quad (7)$$

où k_1 et k_2 sont les coordonnées du centre du cercle K , et r son rayon. En outre on est dans un cas de figure où le cercle passe par l'origine O , donc $c = 0$.

La méthode des moindres carrés est connue pour avoir une robustesse exécrable dès qu'il existe des mauvais points. Nous avons donc adopté la méthode RanSac [7], qui est connue en photogrammétrie et en vision par ordinateur pour sa grande robustesse et sa rapidité de calcul. Il s'agit d'une méthode d'estimation par consensus. On commence par un échantillonnage aléatoire de 3 points qui permet de déterminer un cercle, puis avec les paramètres ainsi calculés, on sélectionne les points dont la distance au cercle est inférieure au seuil prédéfini (t). On cherche à ce stade le meilleur cercle passant par les points ainsi choisis. En analysant visuellement la représentation graphique des points H extraits, nous retrouvons bien les arcs de cercle évoqués précédemment (cf Figure 5).

Pour ce faire, chaque point qui est retenu pour participer à la détermination d'un cercle est retiré de l'ensemble de segments de départ. Le nombre de points minimal participant au consensus est déterminé comme un pourcentage du nombre total de segments détectés, et ce pourcentage est l'un des paramètres importants de réglage de l'algorithme. Par itérations successives, la totalité des cercles est extraite, et si le réglage des seuils est bien adapté, on obtient uniquement les cercles correspondant aux points de fuite (cf. Figure 6).

■ Estimation de l'incertitude sur le point de fuite calculé

Dans la partie 2 nous avons déterminé la matrice variance-covariance des paramètres d'une droite sachant que l'équation de la droite est paramétrable de la façon suivante :

$$-\sin(\theta) \cdot x + \cos(\theta) \cdot y = \rho \quad (8)$$

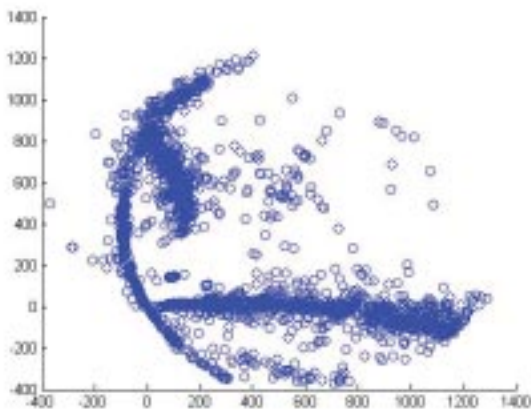


Figure 5 : Exemple des points H . On note qu'ils passent par le point $(0, 0)$ origine de l'image.

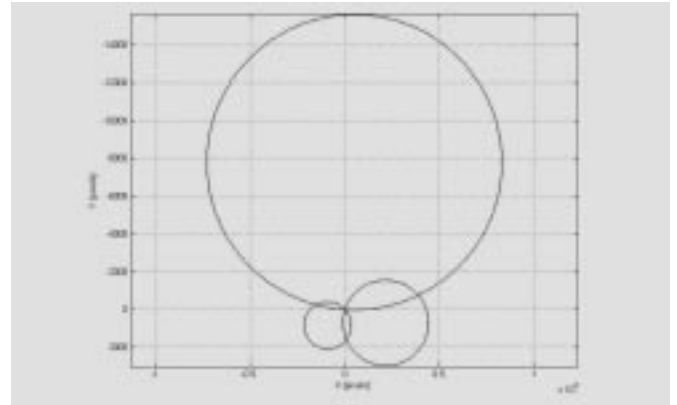


Figure 6 : Cercles extraits à partir des nuages de points H .

Pour chaque segment nous avons calculé les paramètres θ et ρ ainsi que leurs incertitudes, à partir de la matrice variance-covariance :

$$\Sigma_{\theta, \rho} = \begin{bmatrix} \frac{\partial^2}{\partial \theta^2} & \frac{\partial^2}{\partial \theta \partial \rho} \\ \frac{\partial^2}{\partial \theta \partial \rho} & \frac{\partial^2}{\partial \rho^2} \end{bmatrix} \quad (9)$$

Nous cherchons maintenant à propager l'erreur d'estimation des paramètres de la droite sur le pied H de la droite perpendiculaire au segment considéré issue de O . H est défini par :

$$H = \begin{bmatrix} -\rho \cdot \sin(\theta) \\ \rho \cdot \cos(\theta) \end{bmatrix} \quad (10)$$

En utilisant la loi générale de la propagation d'erreurs pour les fonctions non linéaires, et en se basant sur le théorème de Taylor :

$$\Sigma_H = J \Sigma_{\rho, \theta} J^T \quad (11)$$

ou J est la matrice Jacobienne qui contient les dérivées de H selon ρ et θ :

$$J = \begin{bmatrix} \frac{\partial H_x}{\partial \rho} & \frac{\partial H_x}{\partial \theta} \\ \frac{\partial H_y}{\partial \rho} & \frac{\partial H_y}{\partial \theta} \end{bmatrix} = \begin{bmatrix} -\sin(\theta) & -\rho \cdot \cos(\theta) \\ \cos(\theta) & -\rho \cdot \sin(\theta) \end{bmatrix} \quad (12)$$

A partir de (11) nous obtenons la matrice variance-covariance sur le point H :

$$\Sigma_H = \begin{bmatrix} -\sin(\theta) & -\rho \cdot \cos(\theta) \\ \cos(\theta) & -\rho \cdot \sin(\theta) \end{bmatrix} \times \begin{bmatrix} \frac{\partial^2}{\partial \theta^2} & \frac{\partial^2}{\partial \theta \partial \rho} \\ \frac{\partial^2}{\partial \theta \partial \rho} & \frac{\partial^2}{\partial \rho^2} \end{bmatrix} \times \begin{bmatrix} -\sin(\theta) & -\rho \cdot \cos(\theta) \\ \cos(\theta) & -\rho \cdot \sin(\theta) \end{bmatrix}^T \quad (13)$$

En d'autres termes nous disposons du poids attribué pour chaque point H de l'image. Nous nous rapprochons donc du but final c'est à dire la propagation de l'erreur de ρ et θ sur le point de fuite. Rappelons que P , le point de fuite est calculé à partir des paramètres du cercle :

$$P_x = 2 \cdot K_x \quad (14)$$

$$P_y = 2 \cdot K_y \quad (15)$$

En utilisant des moindres carrés pondérés par rapport à H nous

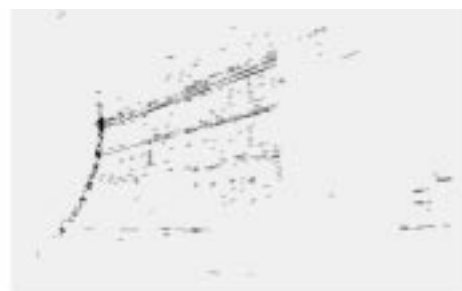
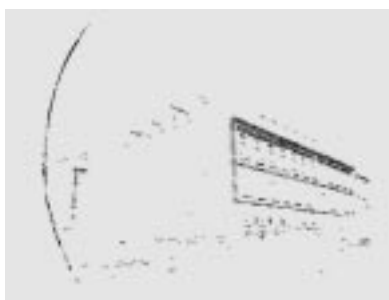


Figure 7 : Les cercles K correspondant aux gerbes perspectives de deux façades d'un bâtiment : on notera les nombreux artefacts correspondant aux parties de l'image situées en dessous de l'horizontale.

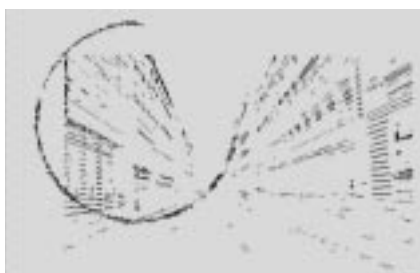
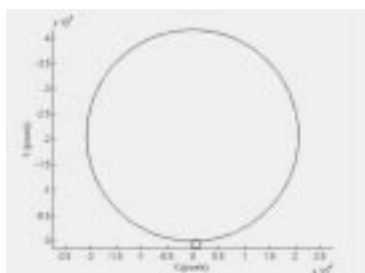
pouvons ainsi obtenir l'incertitude sur les paramètres du cercle, et à partir de celle-ci, calculer la matrice variance covariance du point de fuite, ce qui revient à estimer l'incertitude sur le point de fuite par rapport à la précision du calcul des points H. Dans cette étape un test du χ^2 pourra intervenir pour rejeter les éventuels points H qui ne contribuent pas à ce point de fuite.

■ Implémentation

Notre stratégie pour le calcul de l'incertitude du point de fuite est la suivante : Dans un premier temps et comme cela a été indiqué dans le 3.2, nous ne nous préoccupons que de l'extraction des principaux cercles et du regroupement des segments dans différentes classes. Cette étape accomplie, nous procédons dans chaque groupe de point H aux calculs décrit en 3.3, c'est à dire nous introduisons une incertitude sur nos points H. La dernière étape sera donc de calculer la matrice variance covariance sur le point de fuite correspondant à chaque classe de H.

Annexe

Résultats



Exemple de détection de points de fuite dans une image de rue. De gauche à droite : Image brute, segments extraits, segments verticaux seuls, cercles K extraits (le grand correspond aux verticales, avec un point de fuite très loin de l'image), validation visuelle montrant les segments liés à des horizontales et le cercle K correspondant, image d'origine avec superposition des lignes de fuite extraites.

Validation visuelle de l'algorithme

Une fois que l'on a calculé les paramètres de tous les cercles, pour pouvoir valider l'algorithme, nous traitons le problème en sens inverse, c'est à dire que partant de l'ensemble de tous les segments nous sélectionnons et représentons sur l'image d'origine les segments correspondant aux cercles calculés. On peut ainsi optimiser le réglage des paramètres du RanSac, et identifier où prennent naissance des artefacts résiduels. Nous avons ainsi identifié sans surprise que dans les paysages urbains courants, avec des images acquises à hauteur d'homme, la partie basse de l'image (en dessous de 1,50 m par exemple) était très riche en segments sans aucun rapport avec les bâtiments (peintures au sol, mobiliers urbains, véhicules en stationnement, ...).

Perspectives et conclusions

Nous avons présenté une nouvelle méthode de détection automatique de points de fuite en se basant sur une géométrie



simple dans l'espace 2D de l'image. Les résultats montrent l'efficacité de cette méthode de calcul, qui en outre est rapide, ce qui permet d'envisager son utilisation en temps réel sur le terrain. Il reste néanmoins quelques points à approfondir dans notre algorithme, comme par exemple le paramétrage automatique de la méthode Ransac selon les différentes variantes d'images. Dans nos travaux futurs nous chercherons à exploiter les points de fuite dans les calculs de compensation par faisceaux, afin en particulier de déterminer les paramètres internes de la camera (focale et centre de perspective).

Remerciements

Nous remercions ici les trois collectivités territoriales : Le Mans Métropole, le Conseil Général de la Sarthe, et Conseil Régional des Pays de Loire, pour leur soutien à cette recherche. Nous remercions également le laboratoire Matis de IGN pour la fourniture des différents outils de programmation.

Références

- [1] A. Almansa, A. Desolneux, S. Vamech, *Vanishing points detection without any a priori information*. IEEE Trans. on PAMI, 25(4):502-507, 2003.
- [2] Barnard S. *Interpreting perspective images*. Artificial Intelligence, vol. 21. 1983.
- [3] B. Brillault, B. O'Mahoney, *New method for vanishing point detection*. CVG-IP, Image Understanding, 54(2) : 289-300, 1991.
- [4] C. Bräuer, Burchardt, Klauss Voss, *Robust Vanishing Point Determination in Noisy Images*. Internal report, Digital Image Processing Group. Institute for Compute Science. University of Jena. Germany. 1999.
- [5] R. Deriche, R. Vaillant, O. Faugeras. *From Noisy Edges Points to 3D Reconstruction of a Scene : A Robust Approach and Its Uncertainty Analysis*. World Scientific Series in Machine Perception and Artificial Intelligence , Vol. 2, p. 71-79. 1992.
- [6] R. Deriche, *Using Canny's criteria to derive an optimal edge detector recursively implemented*, Int. J. Computer Vision, Vol. 2, p. 15-20, Avril 1987.
- [7] M. A Fischler, R. C. Bolles, *Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography*. Communications of the ACM, Vol. 24(6):, p.381-395. 1981.
- [8] W. Gander, G. H. Golub, and R. Strebel, *Fitting of circles and ellipses least squares solution*, Technical Report 217, Institut fur Wissenschaftliches Rechnen, ETH Zurich, June 1994.
- [9] F.A Heuvel., *Vanishing point detection for architectural photogrammetry*. International Archives of Photogrammetry and Remote Sensing Vol. XXXII part 5, p. 652-659. 1998.
- [10] E. Lutton, H. Maitre., J. Lopez-Krahe, *Contribution to the determination of vanishing points using Hough transform*. IEEE Trans. PAMI. Vol. 16, N° 4, p.430-438, Avril 1994.
- [11] M. J. Magee, J. K. Aggarwal, *Determining vanishing points from perspective images*. CVGIP, 26(2): p.256-267. 1984.

[12] N. Papanoditis *Thèse Reconstruction 3D de paysages urbains en imagerie stéréoscopique satellitale haute résolution*, Thèse, Université de Nice-Sophia Antipolis, 1998.

[13] L. Quan, R. Mohr, *Determining perspective structures using hierarchical Hough transform*. Pattern Recognition Letters Vol. 9, pp. 279-286. 1989.

[14] T. Tuytelaars, L. Van Gool, M. Proesmans, T. Moons, *The cascaded Hough transform as an aid in aerial image interpretation*. Proceedings of ICCV, p.67-72, 1998.

Contacts

Mahzad KALANTARI

Doctorante à l'ESGT le Mans
associée à l'équipe Image Vidéo Communications
Institut Recherche Communications Cybernétique de Nantes (IRCCyN)
mahzad.kalantari@esgt.cnam.fr

Franck JUNG

Directeur de l'Ecole Supérieure des Géomètres et Topographes (ESGT)
franck.jung@esgt.cnam

ABSTRACT

In architectural photogrammetry as well as in the programs of 3D urban surveys that currently take more and more importance, we search for the solutions allowing to facilitate the fieldwork and the aerotriangulation in the most automatic possible way. In this kind of works it is clear that one can take benefit from the perspectives formed by the parallel lines of the imaged object and of the resulting vanishing points. But for an efficient operation these vanishing points must be recovered without human intervention, problem that up to here has not been solved in a fully satisfactory way. Following our works, we present here a new completely automatic method of detection of vanishing points, robust and of reduced complexity compared to the classic algorithms, in two steps : a detection of segments by a modified algorithm of Canny-Deriche, followed by a characterization of each of the segments by their belonging to a specific circle of the original space. The good performance in terms of robustness is also followed of a complexity reduction since the whole process is achieved in 2D without any external help.