

一种新的消失点检测算法

陈付幸 王润生

(国防科学技术大学 ATR 国家重点实验室 长沙 410073)

摘要 该文在随机抽样一致性算法基础上,提出了一个基于预检验的随机抽样一致性 Preview model Parameters Evaluation RANSAC(PERANSAC)消失点估计算法:该算法在原始RANSAC算法消失点检验前,加入一个预检验步骤,在保证计算结果精度不变的前提下,过滤掉大量偏差较大的消失点,减少了检验的计算量,大大提高了算法的整体效率。大量的实验结果表明,该算法的计算精度与RANSAC算法精度保持一致,计算速度远高于RANSAC算法。

关键词 消失点, Robust, 随机抽样一致性算法, 预检验的随机抽样一致性算法

中图分类号: TP391

文献标识码: A

文章编号: 1009-5896(2006)08-1458-05

A New Vanishing Point Detecting Algorithm

Chen Fu-xing Wang Run-sheng

(National Key Laboratory of ATR, National University of Defence Technology, Changsha 410073, China)

Abstract Preview model Parameters Evaluation RANSAC algorithm (PERANSAC) is given in vanishing point detecting. A preview model parameters evaluation selection is added in the RANSAC algorithm. With guaranteeing the same confidence of the solution as RANSAC, a very large number of erroneous vanishing point obtained from contaminated samples are discarded in the preview evaluation selection. The time of evaluating the quality of the vanishing point is reduced. RANSAC efficiency is significantly improved. PERANSAC algorithm is evaluated on real-world images, a significant increase in speed is shown and the solutions are same as RANSAC.

Key words Vanishing point, Robust, RANSAC, PERANSAC

1 引言

空间一组平行直线,在图像平面上所成的像有且仅有一个交点,称为消失点。消失点不同于成像平面上的其它特征点,它蕴含了直线的方向信息,对消失点进行分析可以提供大量场景三维结构和方向信息;估计场景中的消失点可以在很大程度上简化对场景的描述和理解。因此,消失点估计在机器人导航^[1]、目标重建^[2, 3]、相机定标^[4-6]等方面都有广泛的应用,是计算机视觉领域内一个重要的研究课题。

消失点通常由图像上提取的线段估计得到。由于空间直线投影到图像平面和直线提取过程总存在一定的误差,空间一组平行直线投影到图像平面上后,并不能准确地交汇于一点,每个投影线段和消失点总存在一定的偏差;而且,场景中总会存在大量的直线信息,直线提取算法往往会得到大量的线段,而属于同一组空间平行线的线段只有很少一部分,其它都是错误线段,这严重影响消失点的估计精度。因此,排除错误线段的干扰、降低误差的影响,估计出正确的消失点位置是各种算法研究的重点。

常用的消失点检测算法可以分为3大类:第1类是用空间变换技术,把图像上的信息变换到一个有限的空间上去,

例如高斯球变换^[7-9], Hough 变换^[10];经过变换,把有限远的消失点和无限远的消失点变换为等价的,但是这种方法在变换后,减少了线段和消失点的空间位置信息,丢失了线段长度以及线段与消失点的距离等信息,同时,变换方法的计算精度受积累单元精度的影响;第2类直接利用直线的信息,在图像平面上进行消失点的检测^[11-13],由于要对所有可能的直线交叉点进行计算,算法的效率比较低;第3类用统计估计的方法,根据图像上边缘特征点估计直线的参数,根据这些参数估计出消失点^[14],或者利用消失点和边缘特征点构造代价函数^[15],同时估计直线和消失点。这些统计算法在理论上有一定优势,但是在实际应用中,算法的复杂度高、运算效率比较低,应用价值不大。

随机抽样一致性(RANdom SAMple Consensus, RANSAC)^[16]算法是在计算机视觉领域内应用最广泛的Robust 估计算法之一,它能够处理数据错误率超过50%的模型参数估计。Schaffalitzky^[11]在直接利用线段信息进行消失点检测的算法中引入了RANSAC算法。在Rother^[12, 13]的研究中,为了提高算法效率,也引入了RANSAC算法。但是RANSAC算法需要的计算量仍然比较大,计算效率较低。本文中,我们在RANSAC消失点估计算法基础上,提出了一个基于预检验的PERANSAC消失点估计算法,直接用图像

上的线段信息来估计消失点：算法在 RANSAC 算法消失点检验前，加入一个预检验步骤，在保证计算结果精度不变的前提下，过滤掉大量偏差较大的消失点，减少了检验需要的计算量，大大提高了算法的整体效率。大量实验结果表明，PERANSAC 算法的计算精度与 RANSAC 算法精度保持一致，计算速度高于 RANSAC 消失点估计算法。

本文第 2 节讲述了消失点的基本概念；第 3 节给出 PERANSAC 算法的基本思想、算法可以减少的计算量和计算步骤；第 4 节实验对比了 PERANSAC 算法和 RANSAC 算法，给出实验结果；第 5 节对本文进行小结。

2 消失点

射影几何的知识表明：一组平行直线 $L_1, L_2, L_3, \dots, L_n$ 相交，且交于无限远点。若令该无限远点的成像为 \mathbf{vp} ，当这些直线与成像平面平行时， \mathbf{vp} 位于图像上的无限远位置；若这些直线与成像平面不存在平行关系， \mathbf{vp} 位于图像上有限远的位置，可能在图像内部也可能在图像之外，但我们把这类点统称为消失点。图像上的消失点 \mathbf{vp} 对应一组平行直线，亦即图像上过点 \mathbf{vp} 的一组直线一定存在平行关系。

如图 1 所示，立方体经过透视成像之后，相互平行的直线 a, d, h 相交于一个共同的消失点 \mathbf{vp}_z ，相互平行的直线 c, i, f 相交于一个共同的消失点 \mathbf{vp}_x ，相互平行的直线 b, g, e 相交于一个共同的消失点 \mathbf{vp}_y 。

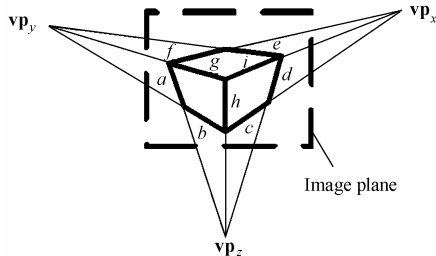


图 1 立方体透视成像形成 3 个方向的消失点
Fig.1 Vanishing points in projective image of cube

在齐次坐标表示下，空间一点 $\mathbf{X} = [X; Y; Z; 1]^T$ ，投影到图像平面上一点 $\mathbf{x} = [u; v; 1]^T$ ；空间一条直线投影到图像平面上的直线段可表示为 $\mathbf{l} = \mathbf{x}_1 \times \mathbf{x}_2$ ， $\mathbf{x}_1, \mathbf{x}_2$ 是线段 \mathbf{l} 的两个端点，消失点可以表示为 $\mathbf{vp} = \mathbf{l}_1 \times \mathbf{l}_2$ 。对一组直线 \mathbf{l}_i ， $i = 1, \dots, m$ ，消失点可以通过求解下面的最小二乘解获得：

$$J(\mathbf{vp}) = \min_{\mathbf{vp}} \sum_{i=1}^m (\mathbf{l}_i^T \mathbf{vp})^2 \quad (1)$$

对表达式(1)，用奇异值分解(Singular Value Decomposition)很容易求得消失点 \mathbf{vp} 。如果消失点 \mathbf{vp} 在直线 \mathbf{l}_i 上，则 $\mathbf{l}_i^T \mathbf{vp} = 0$ ，由于在空间直线投影到图像平面和直线提取过程总存在一定的误差，因此线段和估计出的消失点总存在一定的误差， $\mathbf{l}_i^T \mathbf{vp} \neq 0$ ，称为估计余差。如图 2 (a)为真实场景图像；图 2(b)为图 2(a)一个方向上的直线和用最小二乘求得的消失点。从图 2 中可以看出，直线和估计出的消失点存在偏

差，错误直线对消失点估计影响比较大。

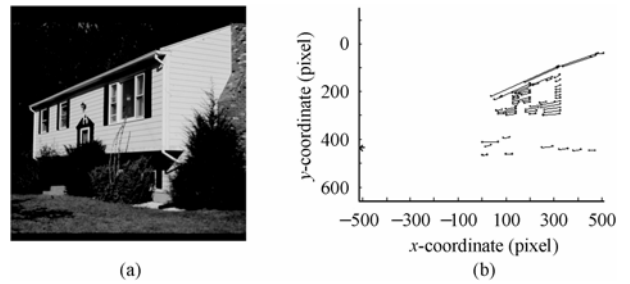


图 2 (a)原始图像 (b) 最小二乘消失点估计

Fig.2 (a) Real image

(b) Vanishing point estimated with least squared criterion

3 PERANSAC 算法

RANSAC 算法采用随机抽样的思想，从随机抽取 M 组样本中找出最优的抽样，并根据最优抽样来选择参与最后计算的原始数据。

算法步骤如下：

(1) 从原始直线段中随机选择 M 组数据样本，每组样本的大小为 3；在一定的置信概率下，保证 M 组样本中至少有一组样本不包含错误直线。

(2) 分别估计 M 组样本的消失点，并用所有直线段来检验消失点(全数据检验)，选择最优的消失点。检验可以用的代价函数有很多，一般采用对应消失点 inliers 数据(与消失点保持一致的直线段)的数量来检验消失点的优劣。

(3) 用最优消失点的所有 inliers 数据来估计最终消失点。

设从原始数据中随机抽取一组直线需要的时间为 t_s ；由一组直线计算消失点需要的时间为 t_c ；用一个直线检验一个消失点需要的平均时间为 t ，则用 N 个直线检验需要的时间为 Nt 。则 RANSAC 算法所需计算时间为

$$T = M(t_s + t_c) + MNt \quad (2)$$

其中 $M(t_s + t_c)$ 为抽取 M 组直线并计算消失点需要的时间， MNt 为 M 个消失点检验需要的时间。

因此，RANSAC 算法的速度主要取决于两个因素：(1) 要选择的样本数 M ，而 M 和数据的错误率、置信概率有关。(2) 消失点检验需要的时间，这和参与检验的消失点数量 M 以及总的的数据量 N 有关。所以，要提高 RANSAC 算法速度，主要从这两个方面入手：想办法减少样本数 M ，或者减少消失点检验需要的时间。

本文通过减少参与全数据检验的消失点数量来减少检验需要的时间，提出 PERANSAC 算法：在对每个消失点检验前，首先进行预检验，如果通过预检验，则进入后续检验，否则这个消失点不再进行后续的全数据检验。算法通过增加样本数量 M ，保证了算法结果和 RANSAC 算法结果的置信概率是一样的，获得的解具有和 RANSAC 算法一样的可信度。算法虽然增加了样本数量 M ，但是，由于大部分消失点都受到错误数据的影响，经过预检验，过滤掉的消失点数量要比增加的样本数量多很多，减少了进行全数据检验的消失点数量，提高了计算效率。

3.1 PERANSAC 算法核心思想

RANSAC 算法在样本集选择的时候要求保证在样本集中至少包含一组样本的数据全是 inliers, M 必须满足下式:

$$1 - (1 - (1 - \varepsilon)^m)^M = P \quad (3)$$

其中 ε 为预期的原始数据错误率, $m = 3$, P 为置信概率, M 为需要抽取的最小样本集数量。由式(3)知道, 一个消失点为好消失点(估计消失点的样本中不包含错误直线段)的概率为 $(1 - \varepsilon)^m$, 在预检验过程中, 好消失点存在的一定概率被误认为是错误的消失点, 被过滤掉, 设 P_f 为好消失点的预检验通过率, 则通过预检验后, 一个消失点为好消失点的概率为 $(1 - \varepsilon)^m P_f$ 。则式(3)变为

$$1 - (1 - (1 - \varepsilon)^m P_f)^{M'} = P \quad (4)$$

用式(4)完全可以描述, 只要抽取 M' 组样本, 就可以在置信概率 P 下, 保证通过预检验后, 保留的消失点中至少有一个好消失点。这保证了经过预处理后, 算法的置信概率与 RANSAC 算法的置信概率是一样的, 保证了算法的准确性。当没有预检验过程时, $P_f = 1$, 对应式(3), 就是 RANSAC 算法。当有预检验过程, $P_f < 1$ 时, $M' > M$, 但由于大部分消失点受到错误直线段的影响, 相对预检验过程可以过滤掉的消失点数量来说, 增加的消失点是很少的。所以, 加入预检验过程后, 在没有改变 RANSAC 算法计算精度的基础上, 减少了计算量, 提高了算法的效率。

3.2 预检验过程

假设进行预检验的样本数为 n 个直线段, 至少存在 n_f 个 inliers 数据的概率为 P_f , 这 n_f 个数据对好消失点是一致的。也就是说, 对好消失点, 在置信概率 P_f 下, 用 n 个数据进行检验时, 至少有 n_f 个数据是 inliers, 如果 inliers 数量少于 n_f , 说明这个消失点不是好消失点, 样本数据中包含有错误直线段, 这个消失点不再参与比较。对好消失点来说, 通过预检验的概率为 P_f 。

$$1 - \sum_{i=0}^{n_f-1} C_n^i \varepsilon^{(n-i)} (1 - \varepsilon)^i = P_f \quad (5)$$

其中 ε 为预期数据错误率, C_n^i 为从 n 个数据中选择 i 个数据的组合; 在实际计算中, 设定 P_f 的最低限, 通过试探不同的 n_f , 计算得到满足条件的最大 n_f 和对应的 P_f 。

对每个消失点, 根据误差项标准偏差来检验直线段是否是消失点的 inliers, 得到消失点对应 inliers 数据的数量, 然后和 n_f 比较, 如果大于 n_f 则该消失点通过预检测, 否则该消失点不能通过预检测。

3.3 PERANSAC 算法减少的计算量

设错误消失点(计算消失点的数据中包含有 outliers)通过预检验的概率为 P_{error} ; 一个消失点预检验需要的时间为 t' 。则 PERANSAC 算法需要的计算时间由两部分组成:

$$T' = T_1' + T_2' \quad (6)$$

其中 T_1' 为抽样和消失点计算需要的时间, T_2' 为消失点预检验和全数据检验需要的时间:

$$T_1' = M'(t_s + t_c) = (M + \Delta M)(t_s + t_c) \quad (7)$$

其中 M 为由式(3)计算得到的抽样数量, $M' = M + \Delta M$ 为由式(4)计算得到的抽样数量;

$$T_2' = M' P_{\text{pass}} N t + M' t' \quad (8)$$

其中 $P_{\text{pass}} = P_f P_{\text{mean}} + P_{\text{error}} (1 - P_{\text{mean}})$, $P_{\text{mean}} = (1 - \varepsilon)^m$ 为抽样中直线全为正确直线的平均概率。 P_{pass} 为消失点通过预检验的平均概率, $P_f P_{\text{mean}}$ 为正确消失点通过预检验的概率, $P_{\text{error}} (1 - P_{\text{mean}})$ 为错误消失点通过预检验的概率。

由式(6)–式(8)得 PERANSAC 算法需要的计算时间:

$$T' = M' (P_{\text{pass}} N t + t' + t_s + t_c) \quad (9)$$

本文 PERANSAC 算法的预检验过程需要时间 $t' = n t$, 式(9)可以写为

$$T' = M' \left((P_{\text{pass}} + n/N) N t + t' + t_s + t_c \right) \quad (10)$$

由式(9)和式(2), 相对 RANSAC 算法, PERANSAC 算法可以减少的计算时间为

$$\Delta T = (M - M' (P_{\text{pass}} + n/N)) N t - \Delta M (t_s + t_c) \quad (11)$$

其中 $(M - M' (P_{\text{pass}} + n/N)) N t$ 的物理意义为经过预检验处理可以减少的消失点检验时间, $\Delta M (t_s + t_c)$ 的物理意义为引入预检验过程增加的抽样和消失点估计时间。

从式(11)可以看出, 影响 ΔT 的主要因素为正确消失点和错误消失点通过预检验的概率 P_f , P_{error} 。 P_{error} 是未知的, 且对不同的原始数据是不同的, 对于相同的原始数据, P_{error} 会随着 P_f 的增加而增加, 且 $P_{\text{error}} \square P_f$ 。

一个好的预检验过程, 要求好的消失点通过率 P_f 高, 而错误消失点通过率 P_{error} 很低。事实上, 当好的消失点通过率 P_f 比较高时, 设置的门限较低, 往往会造成消失点参数通过率 P_{error} 变高, 在进一步的研究中会讨论如何设计更好的预检验过程。

3.4 算法步骤

(1) 由式(5), 根据预检验最小通过概率和数据错误率, 以及参与预检验的样本数 n , 计算 n_f 和通过概率 P_f , 由式(4), 计算需要选择的样本数量 M' ;

(2) 随机抽取 M' 组样本, 并计算每个样本对应的消失点;

(3) 随机选择 n 个数据, 对消失点进行预检验, 重复(3), 直到所有消失点都检验完;

(4) 如果所有的消失点都没有通过预检验, 重新从(1)开始进行计算。如果连续多次出现没有消失点通过预检验, 则调高预期错误率, 重新计算;

(5) 用所有的数据检验通过预检验的消失点, 获得每个消失点的 inliers 数量;

(6) 根据 inliers 数选择最好的消失点, 用该消失点所有的 inliers 数据计算最终消失点。

4 实验结果

由于计算结果带有随机性, 为了更真实地反映算法的特性, 实验给出平均余差和计算时间是 30 次计算结果的平均

值。实验对多幅真实图像一个方向上的消失点进行了求解，分别给出 RANSAC 和 PERANSAC 算法的结果图和两种算法的平均余差和计算需要的时间以及速度提高的百分比。

实验通用条件： $P_f \geq 0.70$ ； $n = 6$ ； $m = 3$ ； $P = 0.95$ 。

4.1 实验 1

求图 2(a)一个方向上直线的消失点；预期数据错误率 $\epsilon = 0.35$ 。如图 3 为 RANSAC 算法和 PERANSAC 算法求得的消失点位置，图中删除了图 2(b)中的错误直线，可以看出，在这两种算法中，错误直线对消失点的估计没有影响；表 1 中分别给出两种算法的平均余差和计算需要的时间，以及速度提高的百分比。

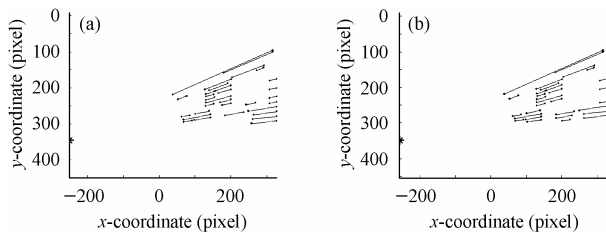


图 3 (a) RANSAC 估计出的消失点位置
(b) PERANSAC 估计出的消失点位置

Fig.3 (a) Vanishing point estimated with RANSAC
(b) Vanishing point estimated with PERANSAC

表 1 平均余差和计算时间
Tab.1 Mean error and time

	平均余差(pixel)	计算时间(s)	速度提高(%)
RANSAC	3.3593	0.1625	14.8
PERANSAC	2.9875	0.1384	

4.2 实验 2

求图 4(a)一个方向上直线的消失点，图 4(b)为图 4(a)中一个方向上的直线；预期数据错误率 $\epsilon = 0.25$ ；图 4(c)和图 4(d)为 RANSAC 算法和 PERANSAC 算法求得的消失点位置，图中只显示正确直线；表 2 中分别给出两种算法的平均余差和计算需要的时间，以及速度提高的百分比。

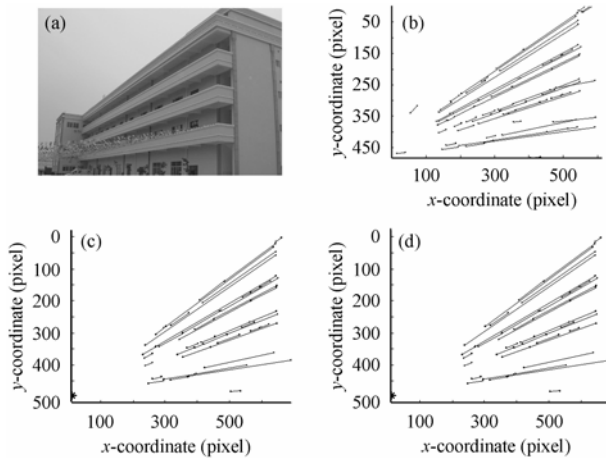


图 4 (a)原始图像 (b) 一个方向上的直线 (c) RANSAC 估计出的消失点位置 (d) PERANSAC 估计出的消失点位置

Fig.4 (a) Real image (b) Lines of one direction
(c) Vanishing point estimated with RANSAC
(d) Vanishing point estimated with PERANSAC

表 2 平均余差和计算时间

Tab.2 Mean error and time

	平均余差(pixel)	计算时间(s)	速度提高(%)
RANSAC	2.6981	0.1596	20.5
PERANSAC	2.6979	0.1268	

4.3 实验 3

求图 5(a)中图像一个方向上直线的消失点，图 5(b)为图 5(a)中一个方向上的直线；预期数据错误率 $\epsilon = 0.35$ ；类似实验 1 和实验 2 给出实验结果：如图 5(c)、图 5(d)和表 3。

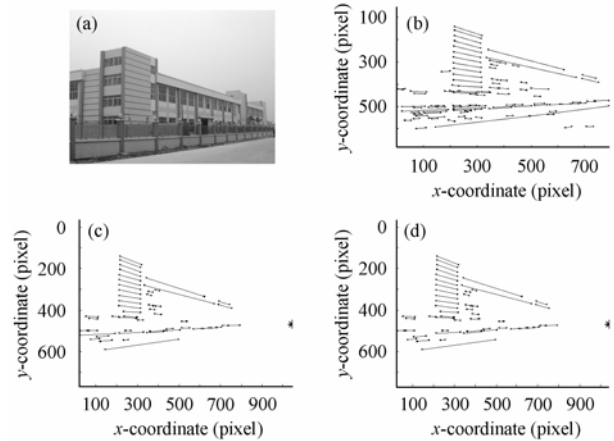


图 5 (a)原始图像 (b) 一个方向上的直线
(c) RANSAC 估计出的消失点位置
(d) PERANSAC 估计出的消失点位置

Fig.5 (a) Real image (b) Lines of one direction
(c) Vanishing point estimated with RANSAC
(d) Vanishing point estimated with PERANSAC

表 3 平均余差和计算时间

Tab.3 Mean error and time

	平均余差(pixel)	计算时间(s)	速度提高(%)
RANSAC	4.0303	0.2100	17
PERANSAC	3.8709	0.1743	

从上述计算结果可以看出，本文算法的计算精度和 RANSAC 算法保持一致，但计算速度有明显提高。

5 结束语

本文分析了以往消失点检测算法及其存在的问题，在原始 RANSAC 消失点估计算法基础上，提出了一个基于消失点预检验的 RANSAC 算法(PERANSAC)，算法在 RANSAC 算法的消失点检验前，加入一个预检验步骤，在不改变计算结果置信度的前提下，过滤掉大量偏差较大的消失点，减少了消失点检验的计算量，大大提高了 RANSAC 算法的整体效率。实验对比了本文算法和 RANSAC 算法，结果证明，本文算法的计算精度和原始 RANSAC 算法精度保持一致，计算速度高于 RANSAC 算法。

参考文献

[1] Straforini M, Coelho C, Campani M. Extraction of vanishing points from images of indoor and outdoor scenes. *Image and*

- Vision Computing*, 1993, 11(2): 91–99.
- [2] Criminisi A, Reid I, Zisserman A. Single view metrology. Proceedings of the 7th International Conference on Computer Vision, Kerkyra, Greece, 1999: 434–441.
- [3] Rother C, Carlsson S. Linear multiview reconstruction and camera recovery. Proceedings of the 8th International Conference on Computer Vision, Vancouver, Canada, 2001, 1: 42–51.
- [4] Svedberg D, Carlsson S. Calibration, pose and novel views from single images of constrained scenes. Proceedings of the 11th Scandinavian Conference on Image Analysis, Kangerlussuaq, Greenland, 1999: 111–117.
- [5] Aprile B C, Using V T. Vanishing points for camera calibration. *International Journal of Computer Vision*, 1990, 4(2): 127–140.
- [6] Liebowitz D, Zisserman A. Combining scene and auto-calibration constraints. Proceedings of the 7th International Conference on Computer Vision, Kerkyra, Greece, 1999: 293–300.
- [7] Quan L, Mohr R. Determining perspective structures using hierarchical Hough transform. *Pattern Recognition Letters*, 1989, 9(4): 279–286.
- [8] Lutton E, Maître H, Lopez-Krahe J. Contribution to the determination of vanishing points using Hough transform. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1994, 16(4): 430–438.
- [9] Barnard S T. Interpreting perspective images. *Artificial Intelligence*, 1983, 21: 435–462.
- [10] Tuytelaars T, *et al.*. The cascaded Hough transform as an aid in aerial image interpretation. Proceedings of the 6th International Conference on Computer Vision, Bombay, India, 1998: 67–72.
- [11] Schaffalitzky F, Zisserman A. Planar grouping for automatic detection of vanishing lines and points. *Image and Vision Computing*, 2000, 18(9): 647–658.
- [12] Rother C. A new approach for vanishing point detection in architectural environments. British Machine Vision Conference (BMVC), Bristol, UK, 2000: 382–391.
- [13] Rother C. Multi-view reconstruction and camera recovery using a real or virtual reference plane. [PHD Thesis], Royal Institute of Technology, 2003.
- [14] McLean C F, Koyyuri D. Vanishing point detection by line clustering. *IEEE Trans. on PAMI*, 1995, 17(11): 1090–1095.
- [15] Minagawa A, Tagawa N. Vanishing point and vanishing line estimation with line clustering. *IEICE Trans. Info. & Syst*, 2000, E83-D(7): 1574–1582.
- [16] Fischler M A, Bolles R C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981, 24(6): 381–395.
- 陈付幸: 男, 1976 年生, 博士生, 研究领域为图像分析与理解、基于图像的三维建模。
- 王润生: 男, 1941 年生, 教授, 博士生导师, 主要研究领域为图像分析与理解、模式识别和信息融合等。