

一种基于特征编组的直线立体匹配全局算法*

文贡坚

(国防科学技术大学 电子科学与工程学院 ATR 重点实验室,湖南 长沙 410073)

A Global Algorithm for Straight Line Stereo Matching Based on Feature Grouping

WEN Gong-Jian

(ATR Key Laboratory, School of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China)

+ Corresponding author: Phn: +86-731-4576343, Fax: +86-731-2586097, E-mail: wengongjian@sina.com

Wen GJ. A global algorithm for straight line stereo matching based on feature grouping. *Journal of Software*, 2006,17(12):2471-2484. <http://www.jos.org.cn/1000-9825/17/2471.htm>

Abstract: Due to line fragmentation, occlusion and projection of conjoint coplanar space straight lines, there are many “one-to-multiple” and even “multiple-to-multiple” mappings between two features sets in the process of stereo matching, but few reliable methods exist to deal with these cases. In this paper, an algorithm based on feature grouping is proposed to solve these problems. Different from the existing approaches, feature grouping is implemented among the feature set which is composed of linear features extracted from two images, and each feature group contains its associated matching relationships. Therefore, stereo matching becomes equivalent to extracting a set of mutually compatible feature groups from the two images. Two major steps involve in the whole matching process. As much putative feature groups as possible are constructed and their match measures are computed by exploiting some viable geometric and photometric constraints, and then a subset of feature groups is searched so that the sum of the associated match measures is the maximum under the condition that any extracted linear feature at most belongs to only a selected feature group. In order to solve the integer optimization problem, a two-stage method is devised. First, the whole problem is divided into many sub-problems. Second, for each sub-problem, a branch-and-bound method is implemented to find the optimal solution. The proposed algorithm is applied to match straight lines extracted from many pairs of real stereo images, and satisfying experimental results are obtained.

Key words: stereo matching; line matching; feature grouping; integer optimization; branch-and-bound method

摘要: 由于直线断裂、遮挡以及共面空间直线投影等因素的影响,从左右图像中提取出来的直线之间会出现“一配多”甚至“多配多”的匹配情况,然而目前很少有算法能可靠地处理这些情况.提出了一种基于特征编组的匹配算法来解决它.与已有方法不同,该方法是在由两幅图像共同组成的直线集上进行编组.每个特征编组包含了内部直线之间的匹配关系.这样,直线匹配的问题就转化为从直线集中提取一些相互兼容的特征编组的问题.整个算法分为两步:首先在几何和辐射约束的前提下构建所有可能的特征编组,并计算每个特征编组的匹配度,

* Supported by the National Natural Science Foundation of China under Grant Nos.60275008, 40571103 (国家自然科学基金)

Received 2005-09-20; Accepted 2006-03-07

然后从所有可能的特征编组中寻找一个特征编组子集,在保证直线集中的每条直线最多属于该子集中一个特征编组的前提下,使得该子集中特征编组的匹配度之和最大.为了解决这个整数规划问题,设计了一种分两阶段的算法:首先将整个问题分为多个子问题,然后对于每个子问题,利用分支定限法寻找最优解.将所提出的算法应用于实际的立体图像对中,取得了满意的结果.

关键词: 立体匹配;直线匹配;特征编组;整数规划;分支定限法

中图法分类号: TP391 文献标识码: A

立体匹配是计算机视觉和摄影测量长期研究的一个基本问题,其目的是匹配从两个不同视点图像中提取出来的特征集,以恢复目标的三维结构^[1].直线是一种重要的特征,目前已有许多研究致力于解决它的一致性问题.直线匹配中主要存在3个问题:(1) 受噪声、照明条件以及提取算法的影响,一条长直线被分割成几段;(2) 一条空间直线在某个视点图像中的投影被部分或全部遮挡;(3) 如果一组空间直线共面且该空间平面通过一幅图像的摄影中心,那么,这些直线在该图像上的投影落在同一条直线;而在另一图像中的投影是由不同方向或偏移量的直线组成.这样,两个直线集之间就会出现“一配零”、“一配一”、“一配多”甚至“多配多”的匹配情况.

在现有文献中,绝大多数算法假定特征之间是“一配一”的关系,它们分两步完成匹配^[2]:首先采用局部算法,依据包括极线几何^[3-5]、几何属性^[6-13]以及辐射相似性^[3,13-17]等在内的一元匹配约束去搜索所有可能的特征匹配对,并计算特征之间的一致性测度;然后依据二元匹配约束准则,如唯一性^[2]、连贯性^[7,18]、最小视差变化^[8]、拓扑约束^[6]、有序性^[9,18]等,采用全局优化的算法来获得最优的匹配关系.这些优化算法包括图搜索^[6,19]、动态规划^[18,20,21]、线性和凹面规划^[22]、贪婪算法^[12]、模拟退火算法^[5]、松弛和一致性标记算法^[8,23]以及神经网络^[14]等.另外,许多算法利用层次计算的策略,以获得更准确而可靠的匹配结果^[21,23-25].

为解决直线断裂的问题,一种途径是匹配之前先对直线进行编组^[6,26].这种思路的优点是直线编组之间的匹配有更多的几何信息可利用来消除模糊性,它的缺点在于,单幅图像对直线编组会引出一些错误的结果;另一种途径是算法设计时允许“一配多”的情况^[4,5].然而就我们了解到的知识,目前还没有一种方法分析“多配多”的情况.

本文的目的就是要设计一种直线匹配算法,它能够解决直线匹配中存在的“一配多”甚至“多配多”的问题.本文通过引出一种新的特征编组概念,将直线匹配的问题转化为提取一些相互兼容的特征编组问题.本文第1节讨论特征编组的定义以及提取方法.第2节讨论如何提取一些相互兼容的最优特征编组子集.第3节列出整个直线匹配算法.第4节给出实验结果.最后是结论.

1 特征编组的定义与提取方法

本节首先分析空间直线在左右图像中的投影情况,然后引出特征编组的定义,最后设计了一种从左右直线中提取特征编组的方法.

1.1 空间直线在左右图像中的投影情况

为了简化,我们分别称空间直线在左右图像上的投影直线为左右投影直线.下面列出了空间直线在左右图像中投影的5种情况:

情况 1:左右投影直线均被完全提取出来.

如图 1 所示,两条投影直线的两个端点都遵循极线约束,只存在“一配一”的情况(图中虚线表示极线).

情况 2:一条投影直线提取完整,而另一条投影直线提取不完整.

假设左投影直线提取完整,如图 2(a)所示,右投影直线可能是以下情况:(1) 只有一条直线,如图 2(b)所示;(2) 由多条直线组成,如图 2(c)所示.在这种情况下,就会出现“一配一”或“一配多”的情况.

情况 3:左右投影直线均提取不完整.

如图 3 所示,左右投影直线均有可能由一条或多条直线组成.在这种情况下就会出现“一配一”、“一配多”

甚至“多配多”的情况.比如,图 3(a)和图 3(b)之间是“一配一”,图 3(c)和图 3(d)之间是“多配多”.

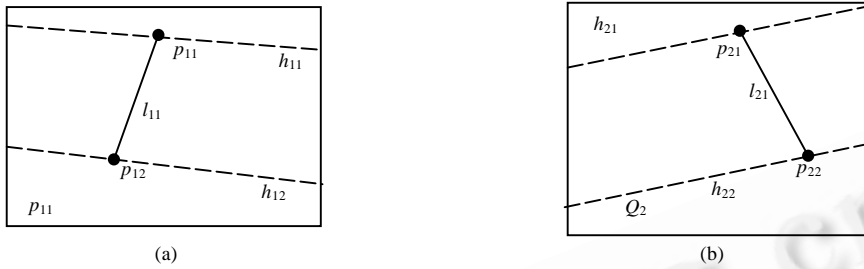


Fig.1 Case 1 of the projection of a space straight line

图 1 空间直线投影情况 1

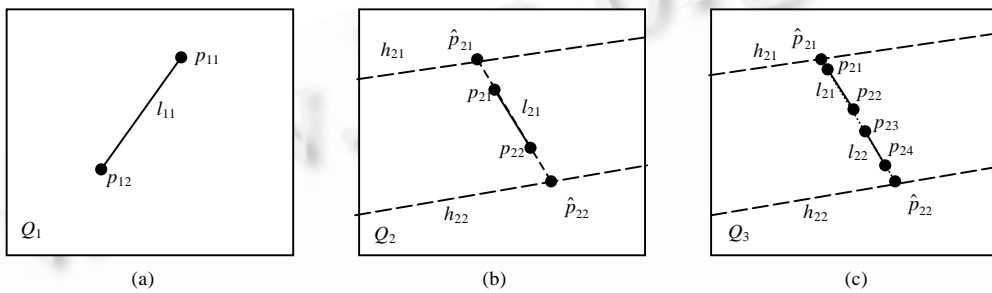


Fig.2 Case 2 of the projection of a space straight line

图 2 空间直线投影情况 2

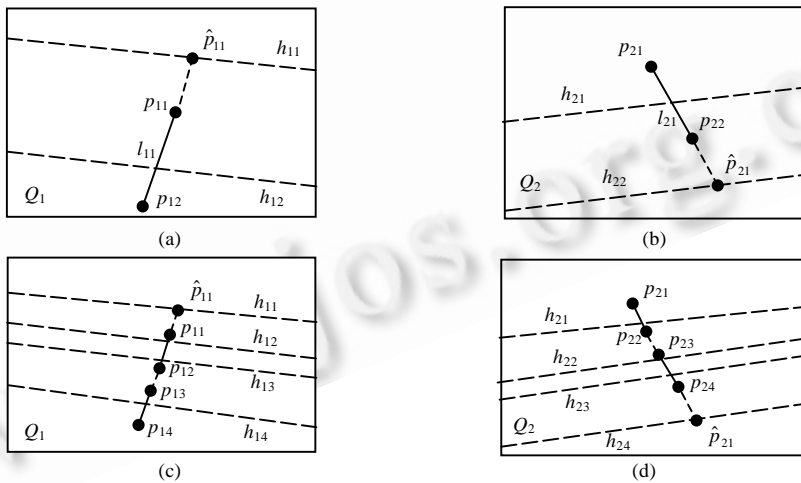


Fig.3 Case 3 of the projection of a space straight line

图 3 空间直线在左右图像中投影情况 3

情况 4:从一幅图像中提取的直线由多条投影直线连接而成.

如图 4 所示,虽然左图像上点 p_{11}, p_{12}, p_{13} 在同一条直线 l_{11} 上,但直线 $p_{11}p_{13}$ 与右图像上直线 l_{21} 是一条空间直线在左右图像上的投影,而直线 $p_{12}p_{13}$ 与右图像上直线 l_{22} 是另外一条空间直线在左右图像上的投影.这样,直线 l_{11} 即由两条左投影直线连接而成.当然,直线 l_{11}, l_{22}, l_{23} 都有可能提取不完整,出现情况 2 和情况 3.

情况 5:只存在一条投影直线或两条投影直线均不存在.

此时,一个或两个相机都观察不到空间直线,造成遮挡,不能恢复该空间直线的位置.这样,一幅图像上提取

出来的直线在另外一幅图像上找不到对应的直线,即“一配零”的情况.

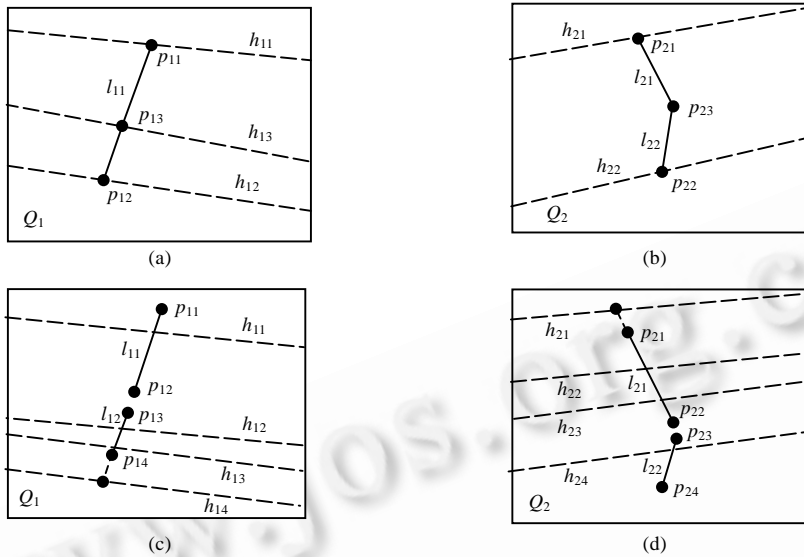


Fig.4 Case 4 of the projection of a space straight line

图 4 空间直线在左右图像中投影情况 4

1.2 特征编组的定义

由上可知,直线的匹配情况可能是“一配零”、“一配一”、“多配一”甚至是“多配多”.因此,要建立左右直线之间的对应性是非常复杂的.换个角度来看,如果我们根据上面列出的几种情况,将左右影像中的有可能匹配成功的特征编在一起,那么,特征匹配的问题就转化为挑选一些合理的特征编组.左右图像特征集之间能够匹配成功的编组就是本文所引出的特征编组的概念.已有文献曾提出过特征编组的概念^[6],但只是在一幅图像上根据相邻特征的相似性编在一起.而我们这里所描述的特征编组,则是将左右影像中有可能匹配成功的特征编在一起,特征编组中已包含了特征的匹配信息.为了简化,我们把从左图像提取的直线称为左直线,从右图像中提取的直线称为右直线.下面给出特征编组的严格定义:一个特征编组是由一条或多条左直线以及一条或多条右直线所组成的一个集合,该集合必须满足如下条件:

- (1) 每一条左直线均能在右直线中找到匹配直线或直线集,而且只能有一种匹配情况,反之亦然;
- (2) 若向该集合添加一条新的左(右)直线,那么必须在该集合中至少找到一条右(左)直线与之匹配.

1.3 特征编组的提取方法

1.3.1 提取直线并计算它们的属性

我们采用基于卡尔曼滤波的直线提取算法从左右图像提取直线^[27],下面计算它们的属性.

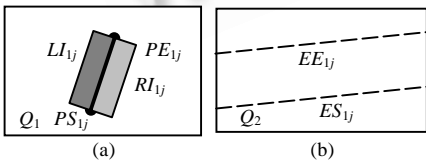


Fig.5 The properties of the left line l_{ij}

图 5 左直线 l_{ij} 的属性

如图 5 所示,假设从左图像 Q_1 中提取出来的 N_1 条直线组成左直线集 $L_1 = \{l_{11}, l_{12}, \dots, l_{1N_1}\}$,从右图像 Q_2 中提取出来 N_2 条直线组成右直线集 $L_2 = \{l_{21}, l_{22}, \dots, l_{2N_2}\}$.对于每条直线 l_{ij} ,根据立体模型以及图像特性计算它的属性,记为 $F_{ij} = \{PS_{ij}, PE_{ij}, \theta_{ij}, ES_{ij}, EE_{ij}, LI_{ij}, RI_{ij}\}$,这里, PS_{ij}, PE_{ij} 表示两个端点,其中 y 坐标偏大的端点为 PS_{ij} ; θ_{ij} 表示直线的方向;如果 $i=1, ES_{ij}, EE_{ij}$ 分别表示端点 PS_{ij}, PE_{ij} 在右图像上的极线,如果 $i=2, ES_{ij}, EE_{ij}$ 分别表示端点 PS_{ij}, PE_{ij} 在左图像上的极线; LI_{ij}, RI_{ij} 分别表示一个固定宽度的图像,该图像的高度大小和方向与该直线的长度和方向一致.图 5 表示左直线 l_{ij} 的属性.

1.3.2 建立匹配矩阵

匹配矩阵表示左右直线集两两之间的匹配关系.我们采用 4 个标准判断两条直线是否匹配:

(1) 应有满足极线几何的匹配点;这个条件是显然的,如果两条直线没有满足极线几何的匹配点,就不能恢复出一条实际存在的空间直线.

(2) 至少有一侧的辐射值是线性相关的;在没有遮挡的情况下,同一物体在左右图像上的辐射灰度值是线性相关的^[26,28].如果空间直线的两侧均没有遮挡,那么,它的两侧区域均在左右图像上成像,则它在左右图像上的投影直线两侧的灰度值是线性相关的;同样,如果空间直线一侧被遮挡,那么,没有被遮挡一侧在左右图像上的投影的灰度值是线性相关的;如果空间直线两侧均被遮挡,这种情况实际上极少发生,本文不予考虑.

(3) 4 个端点的视差值在一定的范围内.

(4) 两直线的方向差小于一个门限值.

应该说,只有前两个条件是必要的.在实际中,为了提高匹配的速度,减少一些不必要的匹配情况,可以引入后两个条件,当然可以放宽门限值来弱化后两个条件.

如图 6 所示,我们描述判断左直线 l_{1j} 与右直线 l_{2i} 是否匹配的过程.

(1) 如果 $|\theta_{1j} - \theta_{2i}| \geq \theta_T$, 则说明它们不匹配,其中, θ_T 为预先设置的一个角度门限值.

(2) 记直线 l_{1j} 与极线 ES_{2i}, EE_{2i} 的交点分别为 LCS_{ji}, LCE_{ji} , 称直线段 $LCS_{ji}LCE_{ji}$ 为直线 l_{2i} 在直线 l_{1j} 上的极线投影,记 LOV_{ji} 为直线段 l_{1j} 和直线段 $LCS_{ji}LCE_{ji}$ 重叠部分的长度;同样可以得到直线 l_{1j} 在直线 l_{2i} 上的极线投影 $RCS_{ji}RCE_{ji}$, 记 ROV_{ji} 为直线段 l_{2i} 和直线段 $RCS_{ji}RCE_{ji}$ 重叠部分的长度.如果 $LOV_{ji} \leq 0$ 或者 $ROV_{ji} \leq 0$, 则说明两条直线不匹配.

(3) 如果直线 l_{1j} 与直线 l_{2i} 匹配,则 $PS_{1j}, PE_{1j}, PS_{2i}$ 和 PE_{2i} 的匹配点分别为 $RCS_{ji}, RCE_{ji}, LCS_{ji}, LCE_{ji}$. 如果它们中有一对匹配点的视差值没有落在预先设置的视差范围 $[D_{min}, D_{max}]$ 内(其中, D_{min}, D_{max} 分别为左右图像的最小最大视差), 则直线 l_{1j} 与直线 l_{2i} 不匹配.

(4) 如图 6 所示, 直线 l_{1j} 和直线 l_{2i} 在极线方向上的重叠区域分别是 $LCS_{ji}PE_{1j}$ 和 $PS_{2i}RCE_{ji}$, 我们分别取出直线 l_{1j} 和直线 l_{2i} 在重叠区域的两侧图像, 并计算它们对应点的灰度差绝对值的平均值, 如果两个平均值都大于预先给定的门限值 G_T , 则说明两条直线不匹配; 否则两条直线是匹配的. 对于一对匹配的直线, 在匹配矩阵中保留 LOV_{ji}, ROV_{ji} 以及 $gdiff_{ji}$ (取其中两个平均值中最小的一个).

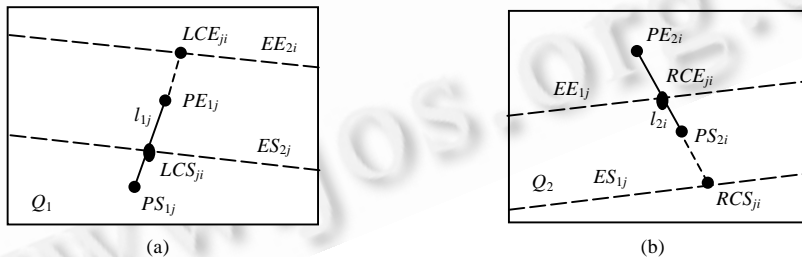


Fig.6 The match relationship between the left line l_{1j} and the right line l_{2i}

图 6 左直线 l_{1j} 与右直线 l_{2i} 的匹配关系

通过上面的处理,我们得到一个 $N_1 \times N_2$ 阶的匹配矩阵, 记为 $M = \begin{bmatrix} m_{11}, \dots, m_{1N_2} \\ \vdots \\ m_{N_11}, \dots, m_{N_1N_2} \end{bmatrix}$, 其中 m_{ji} 表示左直线 l_{1j} 与右

直线 l_{2i} 的匹配情况, 它由 4 个元素组成, 即 $m_{ji} = \{Label_{ji}, Lov_{ji}, Rov_{ji}, gdiff_{ji}\}$. 若直线 l_{1j} 与直线 l_{2i} 匹配, 则 $Label_{ji} = TRUE$; 否则, $Label_{ji} = FALSE$.

1.3.3 寻找每条左(右)直线在右(左)直线集中所有可能的匹配编组

下面,我们以左直线 l_{1j} 为例来寻找它在右直线集中所有可能的匹配情况:

(1) 找出左直线 l_{1j} 在右图像中的所有匹配直线; 取出匹配矩阵 M 的第 j 行, 凡是 $Label_{ji} = TRUE (1 \leq i \leq N_2)$ 的直

线都是 l_{ij} 的匹配直线,将这些直线的编号组成一个集合,记为 $LG=\{no_1, \dots, no_k\}$,其中 k 为匹配直线的数目,称为直线 l_{ij} 的匹配集.

(2) 如果 $k=0$,则说明直线 l_{ij} 在右直线集中没有匹配直线;如果 $k=1$,则说明直线 l_{ij} 在右图像中只有一条匹配直线,也说明直线 l_{ij} 只有一种匹配情况;如果 $k>1$,则进入第(3)步.

(3) 如果 k 条匹配直线中的两条直线可以同时作为直线 l_{ij} 的匹配直线,则称这两条直线对于直线 l_{ij} 是相容

的.我们用一个 k 阶的相容矩阵 $C = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1k} \\ c_{21} & c_{22} & \dots & c_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1} & c_{k2} & \dots & c_{kk} \end{bmatrix}$ 来表示 k 条匹配直线中任意两条直线之间的相容关系,其中, c_{ij}

($1 \leq i \leq k, 1 \leq j \leq k$) 表示直线 l_{2no_i}, l_{2no_j} 之间的相容关系:若两者相容,则 $c_{ij}=1$;否则, $c_{ij}=0$.这样,相容矩阵 C 是一个对角线元素等于 1 的对称矩阵.下面是判断两直线是否相容的准则:

(I) 如果两条直线段共线,且两条直线段没有重叠部分,则两者是相容的,否则不相容;

(II) 如果两条直线不共线,且这两条直线在直线 l_{ij} 上的极线投影直线段没有重叠部分,则两条直线是相容的,否则不相容.

(4) 寻找直线 l_{ij} 的所有可能匹配情况,一种匹配情况就是直线 l_{ij} 的一个匹配编组;作为直线 l_{ij} 的一个匹配编组必须满足如下 3 个条件:

(I) 匹配编组是直线 l_{ij} 匹配集的一个子集;

(II) 匹配编组内的任何两条直线之间是相容的;

(III) 若再往匹配编组内添加任何一条新直线都会破坏它们之间的相容性.

下面给出了寻找匹配编组的算法:先分别以直线 l_{ij} 的匹配集中的每一条直线作为一个匹配编组,并将该匹配编组的标志位置设为 0;然后依次向每个匹配编组内添加一条新直线,如果该直线能够与该匹配编组内任何一条直线相容,则在原有的匹配编组内添加该直线生成一个新的匹配编组,将新编组的标志位置设为 0,将原来编组的标志位置设为 1,这样可以生成所有包含两条直线的匹配编组;同样,再向所有包含两条直线的匹配编组添加新直线,可以生成所有包含 3 条直线的匹配编组.依此下去,构建出所有的匹配编组;最后,所有标志位为 0 的匹配编组就是直线 l_{ij} 所有的匹配编组.

记直线 l_{ij} 所有的匹配编组所组成的集合为 $AMG_{ij} = \{MG_1, MG_2, \dots, MG_{N_{ij}}\}$,其中 N_{ij} 表示直线 l_{ij} 匹配编组的数目; $MG_i (1 \leq i \leq N_{ij})$ 表示一个匹配编组,它保存编组内每条直线在直线集中的编号,编号按由小到大顺序排列.

1.3.4 从左右图像中提取所有可能的特征编组

在讨论算法之前,先讨论提取包含一个特定匹配编组的所有特征编组的方法,以左直线 l_{i_1} 中 MG_j 为例来说明.首先以左直线 l_{i_1} 中 MG_j 为根节点建立一棵树,树的每个节点中保留 3 个信息,分别是直线编号、该直线是左右直线的标记以及匹配编组编号.然后,生成子节点:对于父节点匹配编组中每条直线,如果它的一个匹配编组中存在父节点中的直线且有新的直线没有遍历,则生成一个新的子节点,依此下去,直到再也没有新的节点生成为止.最后,从每个叶节点出发,回溯到根节点,每条路径上所包含的匹配编组就构成一个特征编组.

下面举一个例子说明.如图 7 所示,左右图像是经过核线校正,即匹配点的 y 坐标相等,左右直线集每条直线的匹配编组的结果列列表 1,含有左直线 l_{11} 中匹配编组 $MG_1=\{1,2,3\}$ 的特征编组树如图 8 所示.从图 8 中可以看出,包含左直线 l_{11} 中匹配编组 $MG_1=\{1,2,3\}$ 的特征编组有 1 个,它是 $g_1=\{l_{11}, l_{12}, l_{13}, l_{21}, l_{22}, l_{23}, l_{26}, l_{27}\}$.

将包含每条左直线中的每个匹配编组的特征编组组合起来,就是从左右图像中提取所有可能的特征编组.其实,一个特征编组中包含多个匹配编组,它可以从中任何一个匹配编组出发得到.因此,对于已经编组过的匹配编组,没有必要重新进行编组.这样,从左右图像中提取所有可能的特征编组的方法是:首先以某个特定匹配编组作为起始点,寻找包含它的所有特征编组,将所有被遍历过的匹配编组置上标志;然后,以一个没有置上标志位的匹配编组作为起始点,重复上述过程,直至所有匹配编组均已置上标志位.

由此,我们可以得到图 8 中的全部特征编组为

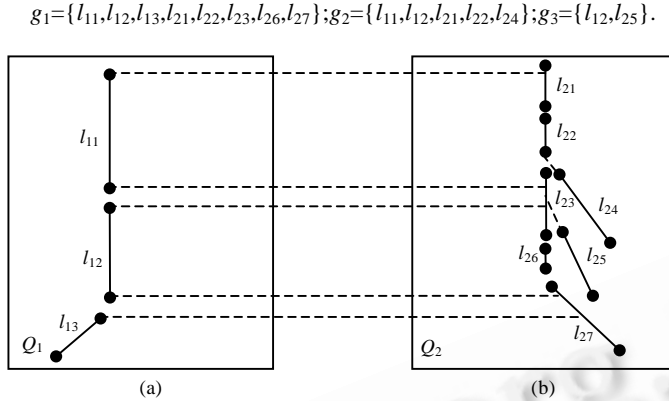


Fig.7 An example for demonstrating the process of extracting feature group

图 7 演示提取特征编组过程的例子

Table 1 Match group set of each line in Fig.7

表 1 图 7 中各直线的匹配编组集

Line	Match group set	Line	Match group set	Line	Match group set
l_{11}	{1,2,3}, {1,2,4}	L_{22}	{1}	L_{26}	{2}
l_{12}	{3,6,7}, {4}, {5}	L_{23}	{1,2}	L_{27}	{2,3}
l_{13}	{7}	L_{24}	{1,2}		
L_{21}	{1}	L_{25}	{2}		

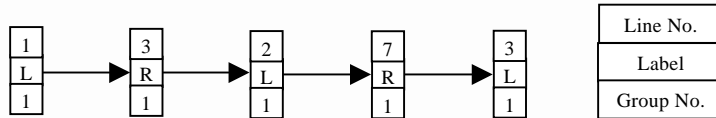


Fig.8 A feature group tree created by the match group $MG_1=\{1,2,3\}$ in the left line l_{11}

图 8 以左直线 l_{11} 中的匹配编组 $MG_1=\{1,2,3\}$ 为根节点建立特征编组树

2 寻找最优的特征编组子集

根据上面的方法可以从左右直线集中提取所有可能的特征编组,也就是左右直线之间所有可能的匹配情况.而此时左右直线中任何一条直线有可能出现在多个特征编组中,也就是说,一条直线可能有多种匹配结果.实际上,一条直线最多允许有一种匹配情况,即左右直线集中的每条直线最多存在于一个特征编组中.本节将讨论如何从所有可能的特征编组中选择合适的特征编组,使得满足上述条件.

2.1 问题的实质

对于每个特征编组,可以定义一个量来衡量其内部各直线之间的匹配程度.这个量越大,表示该特征编组内部各直线匹配越好,称这个量为特征编组的能量值.这样,直线匹配的实质就是从所有可能的特征编组集中选取一些合适的特征编组,在左右直线集中每一条直线只在选取的特征编组中最多出现一次的约束下,它们的能量和最大.

下面对上述问题数学化.记从左右图像提取出来的直线集为 $V=\{L_1 \cup L_2\}=\{v_i | 1 \leq i \leq N, N=N_1+N_2\}$,其中右直线的编号在原有编号上加上 N_1 .假设通过上节描述的方法在直线集 V 中提取了 K 个特征编组,记为 $G=\{g_i | g_i=(\alpha(i), \Phi(i)), 1 \leq i \leq K\}$,其中 $\alpha(i)$ 为特征编组 g_i 的能量值; $\Phi(i)$ 表示特征编组 g_i 中所包含的直线集, $\Phi(i)=\{\phi_{ij} | \phi_{ij}=l, v_i \in V, 1 \leq j \leq R(i)\}$,其中 $R(i)$ 表示特征编组 g_i 中直线的数目, $\Phi(i)$ 中的元素按由小到大顺序排列.我们设函数

$$\delta_{\phi(i)} = \begin{cases} 1, & \text{若特征编组 } g_i \text{ 被选取} \\ 0, & \text{否则} \end{cases} \quad (1)$$

则直线匹配问题就转化为

$$\max \left(\sum_{i=1}^K \delta_{\phi(i)} \omega(i) \right) \quad (2)$$

在下列条件的约束下,

$$\sum_{\substack{i=1 \\ l \in \phi(i)}}^K \delta_{\phi(i)} \leq 1, \quad 1 \leq l \leq N \quad (3)$$

2.2 特征编组能量值的计算

设直线集 $\phi(i)$ 中数值小于或等于 N_1 的元素个数为 M_1 , 即有 M_1 条左直线, 那么右直线的数目 $M_2 = R(i) - M_1$, 定义特征编组 g_i 的能量值 $\omega(i)$ 为

$$\omega(i) = \sum_{m=1}^{M_1} \sum_{n=1}^{M_2} Match(l_{1\phi_m}, l_{2(\phi_m - N_1)}) \quad (4)$$

其中, $Match(l_{1i}, l_{2j})$ 表示左直线 l_{1i} 与右直线 l_{2j} 的匹配度, 其计算为

$$Match(l_{1i}, l_{2j}) = \begin{cases} 0, & \text{如果 } l_{1i} \text{ 与 } l_{2j} \text{ 不匹配} \\ \frac{LOv_{ij} + ROv_{ij}}{2} \times \frac{1}{\sqrt{2 \times PI \times \sigma_g}} \exp\left(-\frac{gdiff_{ij}^2}{2 \times \sigma_g^2}\right), & \text{否则} \end{cases} \quad (5)$$

由式(5)可知, LOv_{ij}, ROv_{ij} 越大, 即左直线 l_{1i} 与右直线 l_{2j} 满足极线约束的点越多, 匹配度越大; σ_g 表示一个高斯分布的方差, 由第 1.3.2 节可知, 如果两直线匹配, 那么它们至少有一侧是来源于同一区域, 因此, 它们有一侧图像的灰度差值应是服从 0 均值的高斯分布. 这样, 图像的平均灰度差值越少, 两直线的匹配度越大.

2.3 问题的求解

式(2)和式(3)所描述的问题是一个整数规划问题, 可采用分支定限法^[29]来求解. 为了减少计算量, 本文提出了一种方法, 下面先讨论几个概念.

定义 1. 对于一个特征编组集 G , 求解式(2)和式(3)所得到的一个特征编组子集称为它的最优特征编组集, 记为 G_{opt} .

定义 2. 若特征编组 g_i 中每一条直线都不在特征编组 g_j 中, 同时, 特征编组 g_j 中每一条直线也不在特征编组 g_i 中, 那么称两个特征编组 g_i 与 g_j 是相容的.

定理 1. 最佳特征编组集 G_{opt} 中任何两个特征编组都是相容的;

证明: 设 g_i 和 g_j 是从最佳特征编组中挑选出来的任意两个特征编组, 假设它们是不相容的, 则特征编组 g_i 中存在一条直线在特征编组 g_j 中, 或者特征编组 g_j 中存在一条直线在特征编组 g_i 中. 无论是哪种情况, 至少有一条直线同时包含在两个特征编组中, 这与式(3)是矛盾的.

定义 3. 若特征编组集 G_i 的任何一个特征编组与特征编组集 G_j 中任何一个特征编组是相容的, 则两个特征编组集 G_i 与 G_j 是相容的.

定理 2. 若一个特征编组集 G 是由多个相容的特征编组子集组成, 那么, 它的最优特征编组集 G_{opt} 也由这些特征编组子集的最优特征编组集组成.

证明: 假设将特征编组集 G 分解成 M 个相互相容的特征编组子集 $G_i = \{u_{ij} \mid u_{ij} = l, g_l \in G, 1 \leq i \leq M, 1 \leq j \leq Q(i)\}$, 其中, $Q(i)$ 表示子集 G_i 中所包含特征编组的数目. 由相容特征编组集的定义可知: 当 $i \neq j (1 \leq i, j \leq M)$ 时, 有

$$\begin{cases} G_i \cap G_j = NULL \\ \left(\bigcup_{l \in G_i} \phi(l) \right) \cap \left(\bigcup_{n \in G_j} \phi(n) \right) = NULL \end{cases} \quad (6)$$

其中, $NULL$ 表示 \emptyset . 这样

$$\max \left(\sum_{i=1}^K \delta_{\phi(i)} \omega(i) \right) = \max \left(\sum_{i=1}^M \sum_{j=1}^{Q(i)} \delta_{\phi(u_{ij})} \omega(u_{ij}) \right) = \sum_{i=1}^K \max \left(\sum_{j=1}^{Q(i)} \delta_{\phi(u_{ij})} \omega(u_{ij}) \right) \quad (7)$$

由定理 2, 我们得到一种求解特征编组集 G 的最优特征编组集的方法, 该方法分为两步: 首先, 将特征编组集尽可能多地分成多个相容特征编组子集; 然后, 对每个特征编组子集求解最优编组集, 将这些最优编组集组合即可得到特征编组集 G 的最优特征编组.

2.3.1 将特征编组集 G 分解成多个相容的特征编组子集

从特征编组集 G 中选取一个没有标记的特征编组作为种子, 将其标记置上, 将其他没有标记且包含有该特征编组直线的特征编组填充过来, 并置上标记, 组建一个特征编组子集. 依此类推, 就可得到一个特征编组子集. 同样, 再以特征编组集 G 选取一个没有标记的特征编组作为新的种子获得另外一个特征编组子集. 依此类推, 直到特征编组集 G 中每一个特征编组均已置上标记, 得到的特征编组子集就是特征编组集 G 的分解结果.

2.3.2 求解一个特征编组子集的最优特征编组集

采用分支定限法求解一个特征编组子集的最优特征编组集. 其实分支定限法就是建立一棵树的过程, 本文中, 树的每个节点包含 4 个元素, 它们是: (1) 该节点的特征编组编号 B_g ; (2) 与特征编组 g_{B_g} 相容的其他特征编组编号所组成的集合 C_{B_g} , C_{B_g} 中的元素按照它对应特征编组的能量值由大到小排列; (3) 从该节点出发到根节点路径上所经过节点的特征编组的能量和 $Energy$; (4) C_{B_g} 中所有特征编组能量和与 $Energy$ 的总和 E_{total} . 建立树的具体过程如下:

(1) 初始化:

i) 建立根节点 R , 根节点 R 中的 $B_g = -1$, C_{B_g} 为特征编组子集所有特征编组的编号, $Energy = 0$, E_{total} 等于特征编组子集所有特征编组能量值和;

ii) 建立一个堆栈 S , 并将根节点 R 压入堆栈;

iii) 设置一个变量 $\max E$ 记录最大能量值, $\max E = 0$; 一个变量 $\max P$ 记录最优节点, $\max P = NULL$.

(2) 判断是否结束:

i) 若堆栈 S 为空, 则进入 (4);

ii) 从堆栈中弹出一个节点 P , 若该节点的 E_{total} 小于 $\max E$ 或者 C_{B_g} 为空, 则返回 i), 否则进入第 (3) 步;

(3) 对节点 P 进行分支:

i) 节点 P 中 C_{B_g} 内所有元素的标记清除;

ii) 根据节点 P 中 C_{B_g} 的第 1 个没有置上标记的元素 s 建立一个新节点 C 作为节点 P 的子节点, 该节点的 $B_g = s$, C_{B_g} 为节点 P 的 C_{B_g} 中所有与 g_s 相容的特征编组编号集合, $Energy$ 等于节点 P 的 $Energy$ 加上 g_s 的能量, E_{total} 等于 C_{B_g} 所有特征编组能量和加上 $Energy$, 如果节点 C 的 $Energy$ 大于 $\max E$, 则 $\max E = Energy$, $\max P = C$;

iii) 将节点 C 的 C_{B_g} 元素和 s 的标记置上;

iv) 重复步骤 i)、步骤 ii), 直到节点 P 中 C_{B_g} 的所有元素均置上标记;

v) 对上面得到的新节点, 按照由小到大的顺序压入堆栈 S 中;

vi) 返回第 (2) 步.

(4) 从最优节点 $\max P$ 出发, 回溯到根节点, 路径上所经过的节点对应的特征编组组成的特征编组子集就是所求的最优特征编组子集.

3 基于特征编组的直线匹配算法

综上所述, 整个直线匹配算法分 4 步执行:

第 1 步: 按照第 1.3.1 节的方法, 从左右图像中提取直线并计算每条直线的属性;

第 2 步: 按照第 1.3 节的方法提取所有可能的特征编组, 并利用第 2.2 节的方法计算每个特征编组的能量;

从左右直线集中提取出的所有可能的特征编组如图 16 所示,一共有 63 个可能的特征编组.本文方法求得的最优特征编组集由 31 个特征编组组成,它们是 {g0,g1,g8,g3,g5,g6,g7,g22,g41,g58,g48,g10,g15,g12,g20,g28,g35,g16,g32,g23,g33,g39,g47,g51,g54,g53,g55,g56,g60,g61,g62}.最优特征编组集中包含了左右直线集的匹配关系.将匹配结果映射到左右直线集中,其结果分别如图 17、图 18 所示.

00: 0; 01: ; 02: ; 03: 3; 04: ; 05: 10; 06: 6; 07: 2-->4; 08: 5; 09: ; 10: 8; 11: 10; 12: 22; 13: 18; 14: 21; 15: 14-->15-->16; 16: 21; 17: 23; 18: ; 19: ; 20: 14-->15-->16-->25; 21: ; 22: 22-->27; 23: 26-->28; 24: 26; 25: 14-->15-->16-->25-->31; 26: 15-->16-->25; 27: 26-->28; 28: 29; 29: 14-->15-->16-->25-->31; 30: ; 31: 32; 32: 22-->34-->35; 33: ; 34: 22-->35; 35: 22-->35; 36: 36; 37: 22-->34-->27,35; 38: 37; 39: 42; 40: 43; 41: 42; 42: ; 43: ; 44: 44; 45: ; 46: 45; 47: 22-->34-->35; 48: 44; 49: 48; 50: 50; 51: 51; 52: ;

Fig.14 All possible match groups of each left line

图 14 所有左直线的可能匹配编组

00: 0; 01: ; 02: 7; 03: 3; 04: 7; 05: 8; 06: 6; 07: ; 08: 10; 09: ; 10: 5-->11; 11: ; 12: ; 13: ; 14: 15-->20-->29; 15: 15-->25-->29-->20,26; 16: 15-->25-->29-->20,26; 17: ; 18: 13; 19: ; 20: ; 21: 14-->16; 22: 32-->34-->35-->37-->12,47-->22,47; 23: 17; 24: ; 25: 25-->29-->20,26; 26: 23-->24-->27; 27: 22-->37; 28: 23-->27; 29: 28; 30: ; 31: 25-->29; 32: 31; 33: ; 34: 32-->37-->47; 35: 32-->34-->35-->37-->47; 36: 36; 37: 38; 38: ; 39: ; 40: ; 41: ; 42: 39-->41; 43: 40; 44: 44-->48; 45: 46; 46: ; 47: ; 48: 49; 49: ; 50: 50; 51: 51;

Fig.15 All possible match groups of each right line

图 15 所有右直线的可能匹配编组

g0: [L:0,R:0,E:0.1132] g1: [L:3,R:3,E:0.0048] g2: [L:5,R:10,E:0.1492] g3: [L:6,R:6,E:0.0220] g4: [L:7,R:2,E:0.1759] g5: [L:7,R:4,E:0.7603] g6: [L:8,R:5,E:0.0001] g7: [L:10,R:8,E:0.7364] g8: [L:11,R:10,E:0.2709] g9: [L:12,47,R:22,E:0.5080] g10: [L:13,R:18,E:0.4018] g11: [L:14,R:21,E:0.0008] g12: [L:15,R:14,E:0.1932] g13: [L:15,R:15,E:0.9058] g14: [L:15,R:16,E:0.3395] g15: [L:16,R:21,E:0.1367] g16: [L:17,R:23,E:0.0361] g17: [L:20,R:14,E:0.0012] g18: [L:20,26,R:15,E:1.8992] g19: [L:20,26,R:16,E:0.5472] g20: [L:20,26,R:25,E:0.2653] g21: [L:22,47,R:22,E:0.7843] g22: [L:22,R:27,E:0.0299] g23: [L:23,R:26,E:1.7782] g24: [L:23,R:28,E:0.1204] g25: [L:24,R:26,E:0.7375] g26: [L:25,R:14,E:0.0113] g27: [L:25,R:15,E:0.2282] g28: [L:25,R:16,E:0.6617] g29: [L:25,R:25,E:0.0043] g30: [L:25,R:31,E:0.0032] g31: [L:27,R:26,E:0.0526] g32: [L:27,R:28,E:0.0729] g33: [L:28,R:29,E:0.4302] g34: [L:29,R:14,E:0.0009] g35: [L:29,R:15,E:5.1687] g36: [L:29,R:16,E:0.0043] g37: [L:29,R:25,E:0.0021] g38: [L:29,R:31,E:0.1835] g39: [L:31,R:32,E:0.3652] g40: [L:32,R:22,E:1.4143] g41: [L:32,R:34,E:0.7198] g42: [L:32,R:35,E:1.1635] g43: [L:34,R:22,E:0.5178] g44: [L:34,R:35,E:0.0000] g45: [L:35,R:22,E:0.5958] g46: [L:35,R:35,E:0.0000] g47: [L:36,R:36,E:0.0667] g48: [L:37,R:22,E:5.2433] g49: [L:37,R:34,E:0.1879] g50: [L:37,R:27,35,E:2.6893] g51: [L:38,R:37,E:0.0011] g52: [L:39,R:42,E:0.0668] g53: [L:40,R:43,E:0.2451] g54: [L:41,R:42,E:0.7382] g55: [L:44,R:44,E:0.1043] g56: [L:46,R:45,E:0.1792] g57: [L:47,R:34,E:0.2125] g58: [L:47,R:35,E:0.9250] g59: [L:48,R:44,E:0.0774] g60: [L:49,R:48,E:0.0758] g61: [L:50,R:50,E:0.1184] g62: [L:51,R:51,E:0.0651]

Fig.16 All possible feature groups

图 16 所有可能的特征编组集

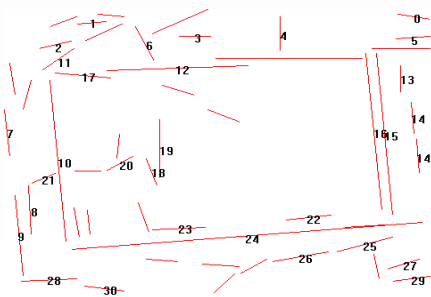


Fig.17 The match result of the lines shown in Fig.11

图 17 图 11 中直线的匹配结果

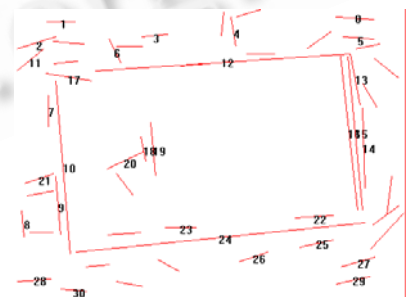


Fig.18 The match result of the lines shown in Fig.12

图 18 图 12 中直线的匹配结果

如图 17、图 18 所示,编号相同的直线为匹配直线,没有编号的直线没有找到匹配直线.例如,第 20 号、第 26 号左直线与第 25 号右直线同为编号 14,则说明第 20 号、第 26 号左直线与第 25 号右直线匹配.在图 17 中,有 32 条左直线找到了匹配直线,21 条没有找到匹配直线,其中 19 条没有匹配直线.第 14 号左直线应该与第 21 号右直线匹配,但由于它提取得不准确,与第 16 号左直线在极线上重叠,故没有找到匹配直线.同样的原因,第 39 号左直线也没有找到匹配直线.在最优特征编组中,第 7 号、第 9 号编组是错误的,即第 22 号左直线与第 27 号右直线、第 47 号左直线与第 35 号右直线并不匹配,它们是由于遮挡或靠近图像边界在另一幅图像中不存在匹

配直线而第 1.3.2 节给出的判断两条直线是否匹配的条件过宽导致的。

图 9、图 10 之间直线匹配的结果列入表 2,表 2 中的真实值是人眼观察得到的。

Table 2 The match result between lines shown in Fig.12 and Fig.13

表 2 图 12、图 13 之间直线匹配的结果

Image	Line number	True match number	Match number by our approach	Correct number	Error number	Miss number
Fig.11	53	32	32	30	2	2
Fig.12	52	31	31	29	2	2

用一对标准的五角大楼立体图像来测试本文的方法.图 19、图 20 分别是两个不同视点拍摄且经核线校正的五角大楼图像.图 21、图 22 表示分别从图 19、图 20 中提取出来的直线(其中长度小于 15 个像素的直线已滤除).最后的匹配结果映射到左右直线集上,其结果如图 23、图 24 所示,编号相同的直线为匹配直线,没有编号的直线说明没有在一幅图像找到匹配直线。



Fig.19 The left image of the pentagon

图 19 五角大楼左图像



Fig.20 The right image of the pentagon

图 20 五角大楼右图像



Fig.21 The left line set from Fig.19

图 21 从图 19 中提取出来的直线

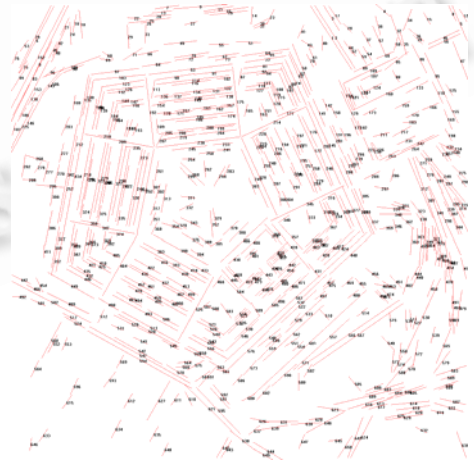


Fig.22 The right line set from Fig.20

图 22 从图 20 中提取出来的直线

两个直线集之间匹配关系的统计结果列入表 3.在表 3 中,绝大多数错误匹配的直线(左直线 46 条、右直线 45 条)是因为在另一幅图像中不存在匹配直线而第 1.3.2 节给出的匹配准则过松而导致的。



Fig.23 The match result of the lines shown in Fig.21

图 23 图 21 中直线匹配的结果

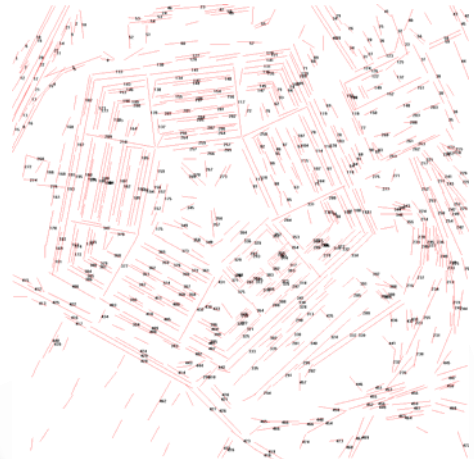


Fig.24 The match result of the lines shown in Fig.22

图 24 图 22 中直线匹配的结果

Table 3 The match result between lines shown in Fig.22 and Fig.23

表 3 图 22、图 23 之间直线匹配的结果

Image	Line number	True match number	Match number by our approach	Correct number	Error number	Miss number
Fig.21	634	467	510	457	53	3
Fig.22	651	461	504	451	53	2

5 结 论

本文通过引入一个全新的特征编组概念来解决直线的立体匹配问题,提出了一种稳健、可靠的直线立体匹配算法.本文首先依据极线几何和灰度相似两个约束条件搜索出直线之间的所有匹配情况,然后在全局约束下挑选统一协调的匹配结果.在构建直线匹配情况时,本文充分考虑了直线匹配之间存在的“多配一”甚至“多配多”的问题,这也正是已有算法在设计时没有详尽考虑的问题,因此,本文提出的方法在这一方面要优于已有的方法.在全局求解中,本文采用了改进的整数规划算法,它能保证算法的有效性.另外,提取特征编组的算法中融合了左右图像的信息,可最大限度地弥补二维直线提取的不完整性,这也是已有算法所不具备的一个特性.

通过对大量实际图像的测试,证实了本文所提出方法的稳健性.虽然在实验中出现了一些错误匹配,但它们大多是由于在另一幅图像中不存在匹配直线的原因所导致的.这种错误可以通过增加匹配约束条件或降低匹配门限来减少,但这种方式不是最佳的:一是它虽然能减少一些错误匹配,但同时也会增加排斥正确匹配的危险;二是如何选择合适的门限也是一件非常困难的事情.本文的下一步研究重点是如何利用已匹配结果在拓扑关系上的协调性来剔除错误匹配.

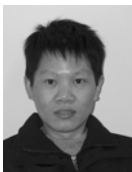
References:

- [1] Dhond UR, Aggarwal JK. Structure from stereo—A review. *IEEE Trans. on Systems Man Cybernet*, 1989,19(6):1489–1510.
- [2] Jones GA. Constraint, optimization, and hierarchy: Reviewing stereoscopic correspondence of complex features. *Computer Vision Image Understanding*, 1997,65(1):57–78.
- [3] Liu Y, Huang T. A linear algorithm for motion estimation using straight line correspondences. *Computer Vision, Graphics, and Image Processing*, 1988,44(1):35–57.
- [4] Schmid C, Zisserman A. The geometry and matching of lines and curves over multiple views. *Int'l Journal Computer Vision*, 2000, 40(3):199–233.
- [5] Cheng YQ, Wang XG, Collins RT, Riseman EM, Hanson AR. Three-Dimensional reconstruction of points and lines with unknown correspondence across images. *Int'l Journal of Computer Vision*, 2001,45(2):129–156.
- [6] Horaud R, Skordas T. Stereo correspondence through feature grouping and maximal cliques. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1989,11(11):1168–1180.

- [7] Medioni G, Nevatia R. Matching images using linear features. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1984, 6(6):675-685.
- [8] Medioni G, Nevatia R. Segment-Based stereo matching. *Computer Vision, Graph and Image Processing*, 1985,31(1):2-18.
- [9] Atalay V, Yilmaz MU. A matching algorithm based on linear features. *Pattern Recognition Letter*, 1989,19(9):857-867.
- [10] Lee CH, Joshi A. On correspondence, line tokens and missing tokens. *Pattern Recognition*, 1995,28(11):1751-1764.
- [11] Bouwmans BT, Dubus JP. A new stereo matching algorithm based on linear features and the fuzzy integral. *Pattern Recognition Letter*, 2001,22(1):133-146.
- [12] Wu M, Leou J. A bipartite matching approach to feature correspondence in stereo vision. *Pattern Recognition Letter*, 1995,16(1):23-31.
- [13] Mcintosh JH, Mutch KM. Matching straight lines. *Computer Vision Graph. Image Processing*, 1988,43(2):386-408.
- [14] Pajares G, Cruz JM. Local stereovision matching through the ADALINE neural network. *Pattern Recognition Letter*, 2001,22(14):1457-1473.
- [15] Henricsson O. The role of color attributes and similarity grouping in 3-D building reconstruction. *Computer Vision and Image Understanding*, 1998,72(2):163-184.
- [16] Nayar SK, Bolle RM. Reflectance based object recognition. *Int'l Journal of Computer Vision*, 1996,17(2):219-240.
- [17] Xie M. Automatic feature matching in uncalibrated stereo vision through the use of color. *Robotics and Autonomous*, 1997,21(3):355-364.
- [18] Li ZN. Stereo correspondence based on line matching in hough space using dynamic programming. *IEEE Trans. on Systems Man Cybernet*, 1994,24(1):144-152.
- [19] Boyer KL, Kak AC. Structural stereopsis for 3-D vision. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1988,10(2):144-166.
- [20] Lee SH, Leou JJ. A dynamic programming approach to line segment matching in stereo vision. *Pattern Recognition*, 1994,27(8):961-986.
- [21] Yip RKK, Ho WP. A multi-level dynamic programming method for stereo line matching. *Pattern Recognition Letters*, 1998,19(9):839-855.
- [22] Maciel J, Costeira JP. A global solution to sparse correspondence problems. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2003,25(2):187-199.
- [23] Wilson RC, Hancock ER. Graph matching with hierarchical discrete relaxation. *Pattern Recognition Letter*, 1999,20(10):1041-1052.
- [24] Venkateswar V, Chellappa R. Hierarchical stereo matching using feature groupings. In: *Proc. of the Int'l Conf. on Tools for AI. San Jose: IEEE Computer Society Press*, 1991. 238-245.
- [25] Marapane SB, Trivedi MM. Multi-Primitive hierarchical (MPH) stereo analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1994,16(3):227-240.
- [26] Chung R, Nevatia R. Use of monocular groupings and occlusion analysis in a hierarchical stereo system. *Computer Vision and Image Understanding*, 1995,62(3):245-268.
- [27] Wen GJ, Wang RS. A robust approach to extracting straight lines. *Journal of Software*, 2001,12(11):1660-1667 (in Chinese with English abstract).
- [28] Luong QT, Fua P, Leclerc YG. The radiometry of multiple images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2002,24(1):19-33.
- [29] Zhang QZ. *Linear Programming*. 2th ed., Wuhan: Wuhan University Press, 2004. 293-305 (in Chinese).

附中中文参考文献:

- [27] 文贡坚,王润生.一种稳健的直线提取算法.软件学报,2001,12(11):1660-1667.
- [29] 张干宗.线性规划.第2版.武汉:武汉大学出版社,2004.293-305.



文贡坚(1972 -),男,湖南宁乡人,博士,副教授,主要研究领域为摄影测量与遥感,图像分析与理解,模式识别,信息融合.