

机器人跟踪打击目标。

3 目标跟踪算法流程设计 (Design of target tracking process)

目前 TLS 系统中限定双方机器人的最快速度为 32 cm/s. 此外由于双方机器人的相对运动, 目标相对的最大移动速度为 64 cm/s. 在对弈过程中双方机器人打击对方的同时需躲避对方攻击, 因此引入躲避策略, 使对手无法预知目标位置. 同时激光接收器的接收面直径为 20 mm. 由以上信息可知图像处理、结果传输、激光打击需在 RH6 机器人的 1 个控制周期 (16 ms) 内完成, 从而保证基于图像信息打击目标的任务能够完成. 因此图像处理时间应尽量缩短. 算法应满足以下条件:

1) 消除图像噪声干扰以实现精确定位, 同时需要压缩数据量以实现快速定位.

2) 将含角点信息的标识置于目标机器人上, 通过识别精确的角点信息计算目标位姿.

3) 首拍在整幅图像中识别出目标的具体位置, 对其余拍图像进行局部搜索以节省时间. 采用动态窗口技术缩小扫描区域^[11], 以本次得到的目标为中心, 以目标在图像平面运动的最高速度和最长采样时间间隔的乘积为半径, 确定一个圆形区域作为目标下一时刻的搜索范围.

根据以上 3 点确定本文图像识别算法的流程, 如图 3 所示.

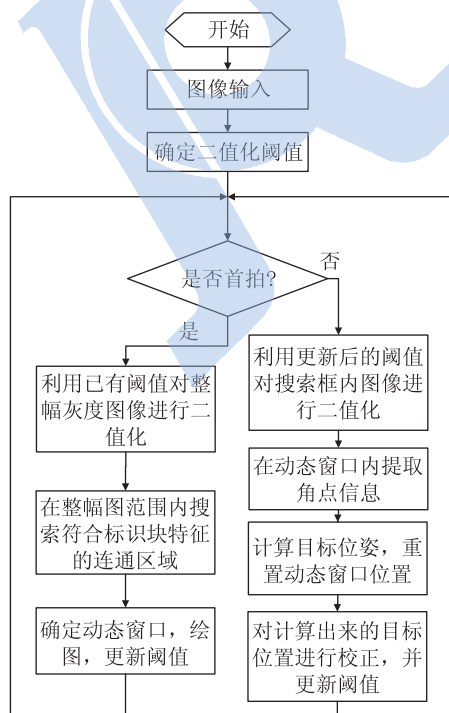


图 3 图像识别算法流程

Fig.3 Image recognition algorithm flow

Shi^[12] 指出“好的目标特征和好的跟踪算法同等重要”. 标识形状应该具有以下几个特点:

1) 标识具有连通特性, 基于灰度及连通特性能较容易地确定其位置, 同时可以抑制噪声干扰.

2) 能利用标识的不对称性确定目标的位姿.

3) 标识本身信息能够较容易识别出来.

本系统标识如图 4 所示. 机器人视觉伺服中, 精确估计目标位姿时一般需要 4 个或更多点. 该标识块的设计在满足以上特点的同时提供了足够的点信息, 具有不对称性, 易于区分各点.



图 4 标识块

Fig.4 Identification block

4 目标跟踪算法改进 (Improvement of target tracking algorithm)

4.1 全局阈值选取

图像分割的效果, 依赖于特定图像以及成像中变化及不变因素 (如噪声和纹理等).

图像分割方法主要有基于阈值、区域生长、聚类法、遗传算法、变形模型分割等. 基于阈值分割实现简单、计算量小、性能较稳定. 特别适用于目标和背景占据不同灰度级范围的图像.

出于实时性考虑, 应尽量减少标识所占像素, 因此一些通过分析直方图的双峰特性来选取阈值的方法不再适用, 如迭代法、直方图凹面法等.

最大类间方差法^[13] 计算简单、稳定有效, 可以投入实时应用. 且当直方图不具备明显双峰时, 也能得到较满意的分割结果. 从模式识别的角度看, 最佳阈值应当使目标类与背景类充分分离, 并采用类别方差来表征该分离性能.

引入类内方差 σ_w^2 、类间方差 σ_b^2 和总体方差 σ_t^2 , 定义 3 个等效的测量准则:

$$\lambda = \frac{\sigma_b^2}{\sigma_w^2}, \quad \kappa = \frac{\sigma_t^2}{\sigma_w^2}, \quad \eta = \frac{\sigma_b^2}{\sigma_t^2} \quad (1)$$

优化第 3 个准则来获取阈值. 实际运用中使用以下简化计算公式:

$$\sigma^2(T) = W_A(\mu_A - \mu)^2 + W_b(\mu_b - \mu)^2 \quad (2)$$

其中, σ^2 为两类间的最大方差, W_A 为 A 类概率, W_b 为 B 类概率, μ_A 为 A 类平均灰度, μ_b 为 B 类平均灰度, μ 为图像总体平均灰度.

利用阈值 T 将图像分成 A、B 两部分, 使 $\sigma^2(T)$ 达到最大值的 T , 即为最佳分割阈值。

为了进一步改善分割效果, 分割前将原图像进行灰度拉伸, 即将原灰度值乘以大于 1 的系数, 使像素间灰度值差别增大, 利于分类。

利用简单统计法 (基于图像统计的阈值选取方法) 和改进的类间方差法分别对不同灯光亮度图像选取阈值并进行二值化的实验, 效果如图 5 ~ 7 所示, 可知采用改进型类间方差法对图像进行分割可以适用于不同光照条件。

4.2 自拍图像的处理方法

对于分割后的图像利用连通特性确定标识位置, 而不采用边缘提取的方法, 可以缩短处理时间。下面针对不同的应用提出两种解决方案。

首先分析处理形状规则标识的方法。由于 DSP 处理器对片内存储器的访问速度是片外存储器的若干倍, 但是片内存储器容量有限, 所以首先利用游程编码对图像进行压缩。

游程编码 (run length coding) 将一行中颜色相同且相邻的像素用计数值和该颜色的颜色值来代替, 再对该颜色和计数值编码。在此基础上, 采用 Blob 方法 (团块搜索法) 提取图像特征。由于 Blob 方法是对图像的整行进行操作及识别, 因此能有效利用游程编码信息, 实现目标的快速定位。



图 5 3 组不同灯光亮度下采集的图像

Fig.5 Images collected under three different light levels



图 6 简单统计法选取阈值

Fig.6 Selecting the threshold with simple statistical method



图 7 改进的最大类间方差法选取阈值

Fig.7 Selecting the threshold with improved Otsu method

利用复杂度算子对不同形状的目标进行区分。复杂度算子用物体区域周长的平方除以物体的面积

表示: $C = P^2/A$ 。该算子与区域大小无关。对于连续图像, 可根据先验知识确定目标的复杂度算子大小, 如圆的复杂度算子为 4π 。

其次对于基于形状不规则、大小不好区分的标识, 本文采用相关系数匹配法替代 Blob 方法。

归一化互相关系数匹配准则公式为

$$R(u, v) = \frac{\sum_{j=1}^w \sum_{k=1}^h (f_{j+u, k+v} - \bar{f})(g_{j, k} - \bar{g})}{\sqrt{\sum_{j=1}^w \sum_{k=1}^h (f_{j+u, k+v} - \bar{f})^2} \sqrt{\sum_{j=1}^w \sum_{k=1}^h (g_{j, k} - \bar{g})^2}} \quad (3)$$

其中, f 为基准图, g 为实时图, $\bar{f} = \frac{1}{wh} \sum_{j=1}^w \sum_{k=1}^h f_{j, k}$, $\bar{g} =$

$$\frac{1}{wh} \sum_{j=1}^w \sum_{k=1}^h g_{j, k}.$$

相关系数满足 $|R(u, v)| \leq 1$, 用来衡量在 $[-1, 1]$ 的绝对尺度内模板与检测区域的相似性。 $R(u, v)$ 越接近 1, 两者越近似地有线性关系^[14]。

$$R(u, v) = \frac{\sum_{j=1}^w \sum_{k=1}^h f_{j+u, k+v} \times g_{j, k} - wh \bar{f} \times \bar{g}}{\sqrt{\sum_{j=1}^w \sum_{k=1}^h f_{j+u, k+v}^2 - wh \bar{f}^2} \sqrt{\sum_{j=1}^w \sum_{k=1}^h g_{j, k}^2 - wh \bar{g}^2}} \quad (4)$$

相对其他匹配准则, 归一化互相关系数对信号强度和等级不敏感的同时有更好的健壮性。但相关系数的求取和归一化操作计算量大, 不适于 DSP 处理器。首先根据文 [15] 对式 (3) 化简, 见式 (4)。

相关系数值的分布具有区域性, 在某些区域会出现相关峰, 同时其余大部分位置的相关系数值都较小。而相关系数匹配法得到的正确位置在相关系数值最大的地方, 即最大相关峰。如果在图中逐点寻找最大相关峰位置, 会导致计算量过大。若能确定最大相关峰所在区域, 再从该区域找出最大相关峰位置, 算法的效率便可以提升。

相关峰之间近似连续过渡, 同时分布具有区域性。通过每隔一定步长计算点的相关系数值, 确定了相关峰的分布情况, 这样就无需对图中的所有点都求取相关系数值。

通过分析, 对算法进行以下改进:

1) 粗、精匹配相结合。粗匹配找出相关系数值最大的区域, 查找过程按一定步长进行, 步长小于模板图片尺寸, 假定步长为 N_{step} , 则计算次数减少为改进前的 $\frac{1}{N_{\text{step}}^2}$ 。

2) 精匹配则在粗匹配选定的区域内进行二次搜

索. 并增加阈值判断, 当精匹配过程中某点的相关系数超过给定门限, 实验中取 0.85, 便停止继续搜索, 确定该像素位置为目标中心.

利用不同角度的模板 (图 8) 对改进后的算法进行验证, 结果见图 9. 该方法不仅对旋转具有一定的适应性, 而且经过以上改进, 提高了算法的时间性能, 匹配时间由 4 s 降为 47 ms.



图 8 不同角度的模板

Fig.8 Templates with different angles

4.3 非首拍图像的处理方法

角点是数字图像中目标的重要局部特征, 通过角点信息能够准确地确定目标位姿. 和其他角点检测算法相比, SUSAN (smallest univalue segment assimilating nucleus) 和 Harris 角点检测算法具有计算简单、位置准确、抗噪声能力强等特点. 本文对 2 种方法进行分析改进, 实现面向不同作业需求的目标识别.

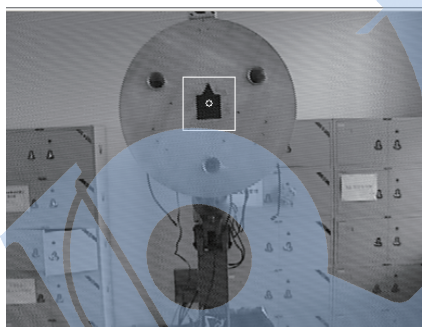


图 9 利用不同角度的模板进行匹配的结果

Fig.9 Matching results of using templates with different angles

4.3.1 改进 SUSAN 方法提取角点信息

SUSAN 方法是由英国牛津大学的 Smith 提出的^[16]. 该方法无需梯度运算, 可保证算法的时间效率满足要求, 提取的特征点分布合理, 具有积分特性, 有较强的抗噪声能力.

SUSAN 算法提取角点的局限性主要表现在:

(1) SUSAN 算法中阈值的选取是关键. 不同图像采用同一阈值提取出来的角点会有很大差异.

(2) 图像灰度分布不均匀以及图像中角点边缘的模糊现象均会影响提取结果. 此外由于光照等因素, 图像会出现部分浅色区域, 使得周围某些灰度值不高的像素, 被误检测成为角点.

(3) 理论上 SUSAN 算法提取出角点的精度与模板大小无关, 模板越大, 计算量也越大, 因此应选取

较小的模板. 但是由于噪声等因素的干扰会造成大量伪角点的产生, 因此模板大小的选取也是 SUSAN 算法实际应用中的难点^[17].

(4) 无论采用的模板大小如何, SUSAN 角点检测算法对图像中的每个像素都要进行多次查表和比较的运算, 因此处理图像时运算量较大.

针对 SUSAN 方法提取角点的不足, 作如下改进:

(1) 精确提取角点需要准确的 SUSAN 阈值, 本文将图像二值化, 这并不影响角点本身的特征. 针对二值图像可以采用固定的 SUSAN 阈值, 同时能够减少边缘模糊及光照等因素对角点定位的影响. 此外图像经过分割后, 可以利用形式简单的比较函数替代原有复杂函数, 减少 CPU 计算量.

(2) 计算各像素的 USAN (univalue segment assimilating nucleus) 值, 当计算完模板的一半时, 便开始判断该核心点的 USAN 值是否已超过 USAN 最大值的一半. 若超过则停止检测该点.

(3) 实际图像一般都包含大量的内部点和背景点, 边缘点只占少部分. 因此, 若 SUSAN 处理只对边缘点进行, 则可提高处理速度. 本文将处理过程分成 2 部分. 第 1 步, 利用 4.1 节得到的图像分割阈值排除背景点; 余下的像素点求以该点为中心 3×3 邻域的灰度平均值, 值为 0 (黑色) 则代表该点为内部点. 以上两种情况不作 SUSAN 处理. 第 2 步, 根据“小模板检测灵敏度高, 耗时短”的特点, 用 5×5 的模板对候选点进行检测, 记录位置; 然后根据“大模板检测准确度高, 误检率低”的优势, 采用 9×9 的模板对记录下的初测角点位置进行二次检测, 筛选出正确角点.

(4) 去除“伪角点”: 判断 USAN 区域的重心和模板中心的连线所经过的像素是否属于 USAN 区域, 属于则角点提取正确^[16].

4.3.2 改进 Harris 方法提取角点信息

Harris 角点检测法基于信号的点特征提取算子^[18], Harris 提取角点的局限性主要表现在:

(1) 进行非极大值抑制确定局部极大值时, 角点提取的结果由阈值决定. 阈值过小会提取出伪角点 (如图 10(a)), 反之则会丢失角点 (如图 10(b)).

(2) 光线扰动、摄像机抖动及图像质量等原因均会导致图像边缘模糊. 由于图像分辨率等因素, 当目标倾斜时, 边缘会产生棱角, 导致“伪角点”的产生 (图 11). (1)、(2) 两种情况对角点定位影响明显.

(3) 高斯平滑中高斯窗口大小的选择会影响处理结果. 若窗口较小, 会因为对噪声抑制不够导致

伪角点的出现;反之,则会因卷积的圆角效应使检测到的角点与实际位置出现偏差。

(4) 用高斯函数对图像作平滑滤波时,高斯方差选取过大会引起过度平滑导致角点丢失^[19]。

(5) 在求局部极大值时的邻域大小会影响提取出角点的数目。

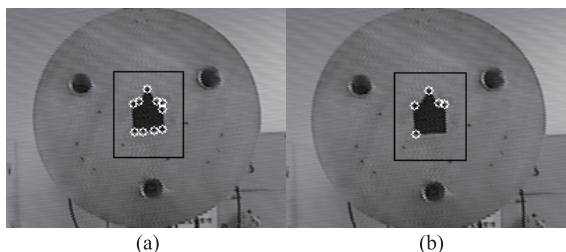


图 10 Harris 检测结果
Fig.10 Results of Harris detection



图 11 伪角点示意图
Fig.11 Schematic diagram of pseudo-corner

针对 Harris 方法提取角点的不足,作如下改进:

(1) Harris 算法中涉及到 4 个参数:阈值、局部极大值的邻域大小、高斯方差和高斯窗口的大小。通常情况下,需要根据不同的图像进行手动调整。文[19]通过实验发现,只有部分图像可以仅通过阈值调整达到角点均匀分布且无角点簇集。本文通过图像分割,使得 4 个参数易于确定。下面对 4 个参数的选取加以分析。

由于是二值图像,因此结合角点特征将阈值固定,并将阈值适当减小以保证能够提取到真正的角点。其次,由于增大极大值邻域可以减少角点簇集现象,同时也不会增加过多的计算量,因此邻域大小可适当放大,本文中设为 9。为了避免角点丢失及位置偏移,本文选取小高斯窗口及方差。

(2) 针对边缘倾斜或噪声等原因产生的“伪角点”,本文先利用 Harris 算子提取出一定数量的特征点,这些特征点都是图像局部范围内的最优特征(包括因参数选取造成的“伪角点”,分布在边缘)。然后利用 SUSAN 算法提取角点的原理对由 Harris 算子检测出的角点进行处理,去除“伪角点”。

如图 11 所示,图中 a 点是应该要找到的角点,而 b 点则是由于倾斜而产生的伪角点。分别计算它们的 USAN 值,与门限值比较(设定为模板面积的 1/4),由此排除了伪角点 b。图 12 为利用改进后的

Harris 方法提取出来的角点。

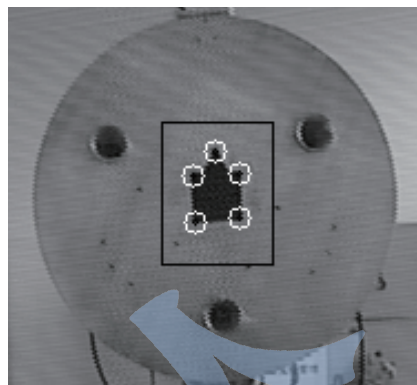


图 12 改进后的 Harris 方法提取角点效果
Fig.12 Corner extraction results with improved Harris method

经过改进,SUSAN 和 Harris 方法的性能均得到改善。针对不同作业需求,可选择适合的方法:当角点较少、实时性要求较高时可采用本文改进后的 SUSAN 方法;目标特征点较多的场合,如利用立体视觉匹配两幅图像时,采用本文改进后的 Harris 方法能够得到较好的结果。

4.4 目标位置的计算、校正及图像分割阈值的更新

对于提取出来的角点,利用其相互位置关系计算最终目标的位置。

单纯的数学计算的方法很难保证计算出来的激光接收器位置的准确性,本文利用模板匹配对目标位置加以校正。

在非首拍处理过程中,需对图像二值化。实际使用中,随着目标及自身移动,最佳分割阈值也随之变化,本文通过直接读取目标标识块的像素值更新阈值,用于下一拍的图像分割。

5 图像处理方法的性能分析及实验验证 (Performance analysis and experimental verification of image processing)

5.1 稳定性能及时间性能的评测及比较

本节利用 2 种能够直接识别激光接收器的方法与本文方法进行实验比较,3 种方法为:

(1) Hough 变换直接搜索激光接收器(圆形,同样采取动态窗口方法)。

(2) 24 邻域链码法直接搜索激光接收器位置。

(3) 识别目标如图 3,每拍进行图像分割,首拍采用 Blob 方法,非首拍采用 SUSAN 方法提取角点,再用相关系数匹配法对计算出来的激光接收器的位置进行校正。

实验中对方机器人沿 6 条轨迹进行运动,利用 3 种方法计算激光接收器位置,并统计错误信息,在

对弈过程中, 双方机器人在打击对方的同时需要躲避攻击, 因此会加入一些躲避策略, 导致图像轨迹无法预知, 于是实际使用中文本对错误拍的处理是使用上一拍的识别结果, 同时将动态搜索窗口的面积适当增大. 实验结果见表 1.

表 1 三种识别方法稳定性比较

Tab.1 Stability comparison of three recognition algorithms

错误 / 帧	方法	Hough 变换	24 邻域链码法	本文方法
跟踪轨迹				
轨迹 1		45/1269	160/1460	30/3901
轨迹 2		36/1569	80/1805	18/4412
轨迹 3		12/1436	49/1652	12/4559
轨迹 4		15/695	30/796	15/2683
轨迹 5		12/775	50/892	12/3007
轨迹 6		15/787	34/906	15/3037
平均出错率 / (次/帧)		0.0205	0.0525	0.0048

上述实验中利用 24 链码法的处理时间为 15 ms, 利用 Hough 变换的处理时间为 30 ms (均不包括滤波、提取边缘等处理).

前两种方法均能通过激光接收器形状特征直接识别目标. 但是由于机器人的大范围运动引起的光照变化及噪声会导致接收器轮廓出现模糊甚至断裂, 因此链码方法无法满足要求. 由于 Hough 变换及类似通过目标形状辨识的方法计算复杂度大且耗时较长, 因此很难满足 TLS 系统对弈任务对实时性的要求.

本文采用具有连通区域的标识, 针对任务对不同方法进行了改进, 能够解决前面 2 类方法所遇到的问题. 本文识别方法的时间性能见图 13, 根据第 3 节总结的不同方法特点和实际情况可以选择不同的方法组合.

对于本文情况, 采用类间方差法确定二值化阈值, 首拍对图像进行游程编码后采用 Blob 方法确定黑色标识块位置. 非首拍采用改进 SUSAN 方法提取角点, 计算 3 个激光接收器的位置后利用模板匹配的方法纠正 3 个激光接收器的位置.

智能摄像机二值化 1 帧图像耗时 2 ms. 本文只对动态窗口内的图像二值化, 其他部分不作处理, 从而使得非首拍二值化时间降到 1 ms 以内.

综上, 识别算法整体运行时间为首拍图像分割耗时 5 ms (阈值计算及二值化), 首拍目标提取 3 ms, 非首拍耗时为 5 ms (二值化、角点提取与目标校正).

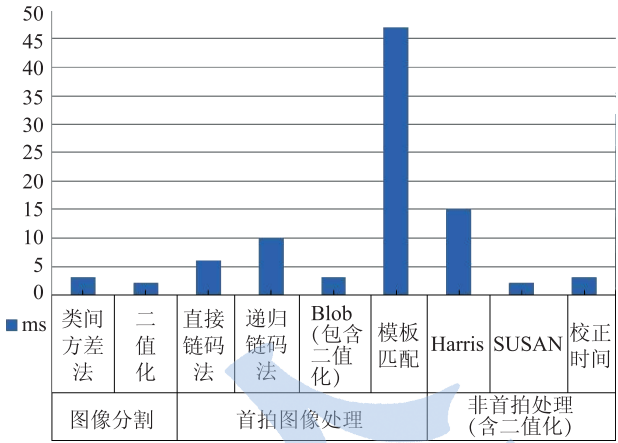


图 13 跟踪算法时间性能

Fig.13 Time performance of the tracking algorithm

5.2 基于图像的激光打击与视觉伺服实验

本节利用 TLS 系统对图像识别方法进行实时性及稳定性验证.

实验 1: 模拟对弈实验中的极端情况, 验证图像识别算法实时性能及定位性能. 控制双方机器人相对运动 (目标相对移动速度最大), 利用本文方法识别目标, 当图像误差为零时控制激光发射器打击目标. 此时通过目标的接收器可以收到激光信号. 当采用表 1 中前两种方法及本文改进前的方法时, 图像识别错误或处理耗时较多, 导致激光无法击中目标或激光打击位置已经越过目标, 从而无法完成对弈任务.

实验 2: 利用 PID 控制器实现系统视觉伺服控制. 如图 14 所示, 平滑的跟踪曲线说明本文方法能够克服双方运动时导致的光照及噪声等因素的干扰, 验证了图像识别算法的稳定性能.

综上本文提出的目标识别方法可以满足 TLS 系统及类似的目标识别需求.

6 结论 (Conclusion)

针对视觉伺服及高速视觉对实时性的要求, 提出了一套能够快速识别并跟踪目标运动的图像处理算法, 此外利用动态窗口提高了系统的实时性. 对于 640×480 像素的 8 位灰度图像, 采集 (图像采集过程与前一拍处理过程并行进行) 后处理 1 幅图像最多需 5 ms (如图 15, 其中 $T_1=4\text{ms}$ 为采集时间, $T_2=5\text{ms}$ 为处理时间), 因此可以应用于控制周期更短的机器人视觉伺服系统. 实验表明, 该图像处理与跟踪算法简单有效, 而且有较好的抗干扰性, 能够满足竞争型机器人对弈任务对实时性能的要求, 也为其他对实时性能要求高的任务提供了一种解决途径.

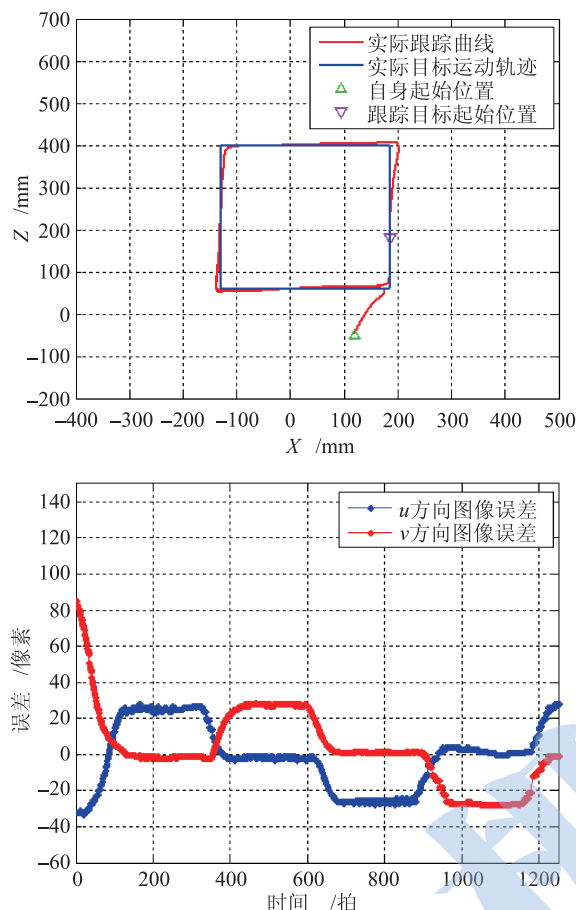


图 14 PID 控制下的轨迹跟踪及图像误差图

Fig.14 Trajectory tracking and image error with PID control

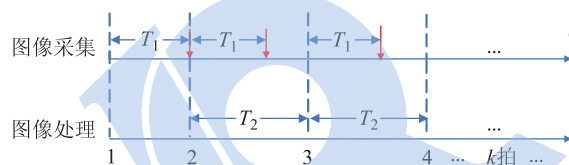


图 15 图像采集与处理时序图

Fig.15 Image collection and processing timing diagram

参考文献 (References)

- [1] 刘景泰, 李海丰, 孙雷, 等. Tele-LightSaber —— 一种高对抗度竞争型网络机器人系统 [J]. 机器人, 2009, 31(6): 505-512.
- [2] Liu J T, Li H F, Sun L, et al. Tele-LightSaber – A kind of competitive networked robots with high degree of opposition[J]. Robot, 2009, 31(6): 505-512.
- [3] Won S H P, Golnaraghi F, Melek W W. A fastening tool tracking system using an IMU and a position sensor with Kalman filters and a fuzzy expert system[J]. IEEE Transactions on Industrial Electronics, 2009, 56(5): 1782-1792.
- [4] Islam M Z, Oh C M, Lee C W. Real time moving object tracking by particle filter[C]//International Symposium on Computer Science and Its Applications. Piscataway, NJ, USA: IEEE, 2008: 347-352.
- [5] 李鹏飞, 喻建平, 李良群. 基于模糊自适应 α - β 滤波的机动目标跟踪 [J]. 系统工程与电子技术, 2008, 30(11): 2138-2141.
- [6] Li P F, Yu J P, Li L Q. Maneuvering target tracking based on fuzzy adaptive α - β filter[J]. Systems Engineering and Electronics, 2008, 30(11): 2138-2141.
- [7] Nakabo Y, Ishikawa M, Toyoda H, et al. 1 ms column parallel vision system and it's application of high speed target tracking[C]//IEEE International Conference on Robotics and Automation. Piscataway, NJ, USA: IEEE, 2000: 650-655.
- [8] Watanabe Y, Komuro T, Ishikawa M. A high-speed vision system for moment-based analysis of numerous objects[C]//IEEE International Conference on Image Processing. Piscataway, NJ, USA: IEEE, 2007: 177-180.
- [9] Senoo T, Namiki A, Ishikawa M. Ball control in high-speed batting motion using hybrid trajectory generator[C]//IEEE International Conference on Robotics and Automation. Piscataway, NJ, USA: IEEE, 2006: 1762-1767.
- [10] Li H F, Liu J T, Li Y, et al. Trajectory planning for visual servoing with some constraints[C]//29th Chinese Control Conference. Piscataway, NJ, USA: IEEE, 2010: 3636-3642.
- [11] Li H F, Liu J T, Li Y. Visual servo of uncalibrated eye-in-hand system with time-delay compensation[C]//8th World Congress on Intelligent Control and Automation. Piscataway, NJ, USA: IEEE, 2010: 1322-1328.
- [12] Vision Components GmbH Ettlingen. VCRT 5.0 software manual[Z]. Germany: Vision Components GmbH Ettlingen, 2007.
- [13] 杨晶. 基于计算机视觉的一体化乒乓球测速方法研究 [D]. 哈尔滨: 哈尔滨工业大学, 2002.
- [14] Yang J. Study on integrated velocity measurement method of ping pong ball based on computer vision[D]. Harbin: Harbin Institute of Technology, 2002.
- [15] Shi J B, Tomasi C. Good features to track[C]//IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE, 1994: 593-600.
- [16] Otsu N. Discriminant and least square threshold selection[C]//4th International Joint Conference on Pattern Recognition. Stevenage, UK: IEE, 1979: 592-596.
- [17] Brown L G. A survey of image registration techniques[J]. ACM Computing Surveys, 1992, 24(4): 325-376.
- [18] Sun C M. Multi-resolution rectangular subregioning stereo matching using fast correlation and dynamic programming techniques[R/OL]. [2010-10-09]. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.31.8269>.
- [19] Smith S M, Brady M J. SUSAN – A new approach to low level image processing[J]. International Journal of Computer Vision, 1997, 23(1): 45-48.
- [20] 张映权, 王琼华, 李大海, 等. 改进的 SUSAN 角点检测算法 [J]. 现代电子技术, 2009, 32(20): 42-44.
- [21] Zhang Y Q, Wang Q H, Li D H, et al. Improved SUSAN corner detection algorithm[J]. Modern Electronic Technique, 2009, 32(20): 42-44.
- [22] Harris C G, Stephens M. A combined corner and edge detector[C]//Proceedings of the 4th Alvey Vision Conference. 1988: 147-151.
- [23] 侯建辉, 林意. 自适应的 Harris 棋盘格角点检测算法 [J]. 计算机工程与设计, 2009, 30(20): 4741-4743.
- [24] Hou J H, Lin Y. Adaptive Harris X-corner detection algorithm[J]. Computer Engineering and Design, 2009, 30(20): 4741-4743.

非完整约束下的机器人运动规划算法

徐 娜¹, 陈 雄¹, 孔庆生¹, 韩建达²

(1. 复旦大学电子工程系, 上海 200433; 2. 中国科学院沈阳自动化研究所, 辽宁 沈阳 110016)

摘 要: 由非完整约束定义提出一种改进 RRT (快速搜索随机树) 算法, 解决受动力学约束的移动机器人运动规划问题. 该算法将移动机器人的非完整约束条件与 RRT 搜索算法相结合. 针对 RRT 算法在全局状态空间均匀随机搜索导致算法无谓耗费代价大的缺陷, 引入目标偏向思想, 并选择计算复杂度低的距离参数提高求解速度. 通过几类典型的非完整约束下的机器人运动规划仿真实验, 证实了该算法的有效性和高效性.

关键词: 运动规划; 非完整约束; 快速搜索随机树 (RRT); 目标偏向; 动力学模型

中图分类号: TP242

文献标识码: A

文章编号: 1002-0446(2011)-06-0666-07

Motion Planning for Robot with Nonholonomic Constraints

XU Na¹, CHEN Xiong¹, KONG Qingsheng¹, HAN Jianda²

(1. Department of Electronic Engineering, Fudan University, Shanghai 200433, China;

2. Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China)

Abstract: An improved RRT (rapidly-exploring random tree) algorithm is presented from the definition of nonholonomic constraints to solve the motion planning problem for mobile robots with dynamics constraints. The algorithm combines the nonholonomic constraints of mobile robot with RRTs. For RRTs, in order to reduce unnecessary costs caused by uniformly random searching in the global state space, the idea of goal bias is introduced and the distance parameter with low computational complexity is also selected to increase the calculation speed. Simulation experiments with several typical robot motion planning with nonholonomic constraints verify the validity and high efficiency of the algorithm.

Keywords: motion planning; nonholonomic constraint; RRT (rapidly-exploring random tree); goal bias; dynamics model

1 引言 (Introduction)

机器人运动规划是移动机器人的基础性问题, 在理论和应用方面都面临许多重要的挑战^[1]. 运动规划与路径规划是两个不同概念. 路径规划是纯几何问题, 是在机器人的位形空间中找到一条从起点到终点的无碰撞路径, 该位形空间只考虑障碍物引起的约束, 并不考虑机器人的动力学约束, 规划出的路径在实际应用中可能根本无法实现^[2]. 运动规划是包括路径规划和运动控制的更为广泛的概念, 运动控制输出速度或力等控制信号, 那么机器人行走速度、旋转角度等都要受控制信号限制, 即要满足动力学约束.

现有规划算法多为路径规划算法, 如人工势场法、遗传算法、蚁群优化算法等. 这些算法不能很好地解决非完整约束下的机器人规划问题, 而且大多需要对空间中障碍物进行确定描述, 计算复杂度与

机器人自由度呈指数关系, 易出现维数灾难.

快速搜索随机树算法 (rapidly-exploring random tree, RRT)^[3] 是基于采样的单查询规划法, 采用对状态空间随机采样的方法把搜索树导向空白区域, 避免对空间的建模, 搜索速度快. RRT 算法属于典型的树结构算法, 树的两个路径节点之间可以选用经过动力学仿真生成的可执行曲线, 能直接应用到非完整约束下的机器人运动规划中. RRT 算法存在一些固有缺陷, 如全局空间中进行均匀的随机搜索导致算法计算量大, 而最近邻 (nearest-neighbor) 算法在解决高自由度机器人规划问题时可能成为制约其速度的瓶颈.

近年来, 研究者提出了许多 RRT 改进算法, 例如为了使随机树易于朝目标点方向生长, 文 [4] 提出目标偏向思想, 文 [5] 借鉴 A* 算法引入启发式代价函数, 两种方式均提高了 RRT 算法的搜索效

率; LaValle 和 Kuffner 提出双向搜索树 (bi-RRT) [4], 从起始点和目标点并行生成两棵 RRT, 直至两棵树相遇, 算法收敛; 文 [6] 又提出 RRT-connect 算法提高节点搜索效率, 该算法只适用于路径规划; 为使 RRT 算法适用于具有狭窄通道的复杂环境, 文 [7] 提出基于障碍物的 RRT 算法, 而文 [8] 则提出利用关键构形的思想予以解决; 针对区域搜索代价不均一的环境, Ferguson 和 Stentz 提出任意时刻 RRT^[9] 算法; 考虑到基本 RRT 算法得到的路径长度一般是最优路径的 1.3 ~ 1.5 倍, Smith 研究变分法技术^[10] 使其达到最优。

本文针对非完整约束下的机器人动力学模型, 提出一种改进的 RRT 运动规划算法, 引入目标偏向思想和距离参数选择机制来解决现有 RRT 算法的缺陷. 通过几类典型案例的比较实验证实了本文算法的有效性 & 高效性。

2 非完整约束 (Nonholonomic constraint)

“非完整”的概念起源于近代分析力学, 最早出现于德国学者 Hertz 的著作 “Die Prinzipien der Mechanik”^[11]. 非完整约束是指含有系统广义坐标导数且不可积的约束. 一般的受动力学约束的受限系统的数学表达式为

$$ds/dt = F(s, u, t) \quad (1)$$

$$C(s, ds/dt, t) = 0 \quad (2)$$

其中, $s \in \mathbb{R}^n$ 为系统状态变量, $u \in \mathbb{R}^m$ 为系统控制变量, t 为时间, $C(s, ds/dt, t)$ 为约束. 如果存在函数 $G(s, t)$ 使得

$$dG(s, t)/dt = C(s, ds/dt, t) \quad (3)$$

成立, 则称系统是完整系统, 对应的约束为完整约束; 反之则称系统是非完整系统, 对应的约束为非完整约束. 非完整约束系统 (简称非完整系统) 包括车辆、移动机器人、某些空间机器人、水下机器人、欠驱动机器人和运动受限机器人等, 因此, 非完整约束下的运动规划研究具有广泛的应用背景和重要的应用价值。

非完整约束是对系统广义坐标导数的约束, 不减少系统的位形自由度, 这使得系统的独立控制个数少于系统的位形自由度. 以 2 维平面上的车型机器人为例, 机器人的位形空间是 3 维的, 包括 2 个位置坐标和 1 个方向角, 因此平面上车型机器人的一个位形可以用 (x, y, θ) 表示, 如图 1 所示^[12], (x, y) 是在系统坐标系下的坐标, θ 是车体纵向主轴与坐

标系之间的夹角. 由于非完整约束条件下, 轮子只能在平面内做纯滚动、无滑动的运动, 在任何运动瞬间车体的速度必然指向机器人的主轴, 即有:

$$dx/dy = \tan \theta \quad (4)$$

展开后得到一个非完整等式约束:

$$-\sin \theta dx + \cos \theta dy = 0 \quad (5)$$

正是这个约束使得机器人在位形空间的可控维数变成 2 维。

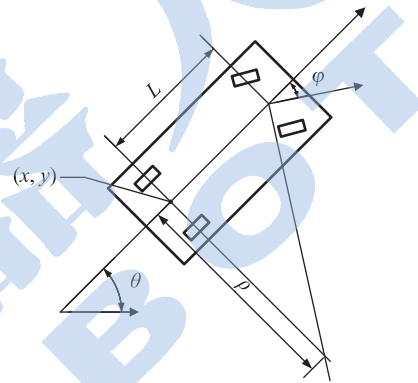


图 1 车型移动机器人运动模型

Fig.1 Motion model of a car-like mobile robot

通过非完整约束的定义及以上分析可以发现, 常规的路径规划算法并不适用于非完整约束下的运动规划问题, 不考虑动力学约束获得的规划路径不一定对应实际工作空间中的可行路径。

3 改进 RRT 算法 (The improved RRT)

3.1 RRT 算法

RRT 算法是一种随机采样的典型树结构算法, 以状态空间中的起始点 s_{int} 为根节点, 通过随机采样逐渐增加叶节点, 生成一个随机扩展树. 当树的节点包含了目标点 s_{goal} 时搜索结束, 反向追溯到根节点即可得到规划路径. 算法基本步骤如下:

```

GENERATE_RRT( $s_{int}, s_{goal}, K, \Delta t$ )
1 T.int( $s_{int}$ );
2 for  $k=1$  to  $K$  do
3    $s_{rand}$  = RANDOM_STATE();
4    $s_{near}$  = NEAREST_NEIGHBOR( $s_{rand}, T$ );
5    $u$  = SELECT_INPUT( $s_{rand}, s_{near}$ );
6    $s_{new}$  = NEW_STATE( $s_{near}, u, \Delta t$ );
7   if COLLISION( $s_{new}$ )
8     continue;
9   T.add_vertex( $s_{new}$ );
10  if  $\|s_{new} - s_{goal}\| \leq d$ 
11    return Reached;
12 Return T

```


其中 s_{int} 、 s_{goal} 表示起始状态和目标状态, 函数 $\text{RANDOM_STATE}()$ 用于在状态空间中随机采样一个点, 然后根据 NEAREST_NEIGHBOR 函数找出树上所有节点中离 s_{rand} 最近的点, 即 s_{near} . 第 5、6 行程序用于求出转换后的状态. 对于非完整约束的机器人系统, 其状态转换与动力学模型相关, 即新节点需要按照动力学方程来计算. 非完整约束方程多为 $ds/dt = f(s, u)$ 的形式, 矢量 u 选自控制输入集合 U , 矢量 s 表示系统的状态. 在给定初始状态即 s_{near} 和控制输入 u 的情况下通过对函数 f 在一段时间区间 Δt 上进行积分便可得到 s_{new} . 本文选用欧拉积分公式 (6). 若对精度要求较高, 则可以选用 Runge-Kutta 积分法.

$$x_{\text{new}} \approx x + f(x, u)\Delta t \quad (6)$$

得到新节点后要判断其是否在障碍物内部, 若是则返回 for 循环处重新选取随机点, 若在自由位形空间内则将新节点加入树中, 直到加入的新节点与目标节点间的距离足够小时搜索结束, 返回规划路径.

3.2 改进 RRT 算法

由 RRT 算法流程分析可知, 一方面随机采样规划方式在全局空间中进行均匀随机搜索, 导致算法计算量大, 造成计算浪费, 代价大; 另一方面, RRT 算法依赖最近邻算法, 在解决高自由度机器人规划问题时可能成为制约其速度的瓶颈. 为克服 RRT 算法的缺陷, 本文引入目标偏向思想和距离参数选择机制. 目标偏向思想就是在选择随机状态时先按均匀概率分布随机获得一个概率值 p , 若其小于设定的阈值, 则 s_{rand} 仍按照函数 $\text{RANDOM_STATE}()$ 随机选取; 若大于阈值, 则令 $s_{\text{rand}} = s_{\text{goal}}$. 即算法步骤中第 3 行处作如下改动:

```

3  p = rand(0 ~ 1)
   if p < threshold
       s_rand = RANDOM_STATE();
   else
       s_rand = s_goal

```

通过引入目标偏向思想减小搜索的随机性降低无谓耗费, 而距离参数选择机制改进原 RRT 算法中 NEAREST_NEIGHBOR 函数距离参数 ρ 的选取. 计算最近节点时需算出树上所有节点到随机点 s_{rand} 的距离 ρ , 原 RRT 选用如式 (7) 的欧氏距离:

$$d_{\text{Euclid}} = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \quad (7)$$

式 (7) 距离的参数选取需两次平方再开方, 无疑会增加计算的复杂度, 本文采用对角线距离求取

ρ . 对角线距离的定义式如下:

$$d_{\text{diagonal}} = \max(|x_a - x_b|, |y_a - y_b|) + (\sqrt{2} - 1) \min(|x_a - x_b|, |y_a - y_b|) \quad (8)$$

与欧氏距离相比, 对角线距离计算简单, 在需要计算大量节点距离时能够提高运算速度.

4 仿真实验 (Simulation)

为验证改进 RRT 算法的有效性和高效性, 本文采用了小汽车、具有双推力器的平板及单轮车等典型的具有二阶约束方程的机器人模型.

4.1 小汽车模型

小汽车是典型的受非完整约束的动力模型, 如图 1 所示. 从驱动机制的角度分析, 许多机器人采用与汽车同样的驱动系统. 这样的系统有两种控制装置: 加速装置和转向驱动轮. 其状态变量为 (x, y, θ, v, ϕ) , 其中位置参考点 (x, y) 是两个后轮轴的中点, θ 表示小汽车的前进方向, ϕ 表示转向角度, v 表示前向速度. 小汽车的控制变量为加速度 u_0 (单位为 m/s^2) 和车轮旋转速度 u_1 (单位为 $^\circ/\text{s}$). 假定前后轮轴之间的距离为 L . 运动约束方程为

$$\begin{pmatrix} dx/dt \\ dy/dt \\ d\theta/dt \\ dv/dt \\ d\phi/dt \end{pmatrix} = \begin{pmatrix} v \cos \theta \\ v \sin \theta \\ v \tan \phi / L \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} u_0 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_1 \quad (9)$$

控制输入 u_0 和 u_1 需要满足一定的限制条件: $|u_0| \leq u_{0,\text{max}}$, $|u_1| \leq u_{1,\text{max}}$. 因为机械约束要求转动角度满足约束 $|\phi| \leq \phi_{\text{max}}$, 本文在仿真过程中设定: $u_{0,\text{max}} = 1 \text{ m/s}^2$, $u_{1,\text{max}} = 100^\circ/\text{s}$, $\phi_{\text{max}} = 50^\circ$, $v_{\text{max}} = 3 \text{ m/s}$, $\Delta t = 0.1 \text{ s}$, $L = 0.5 \text{ m}$.

4.2 带有推力器的平板模型

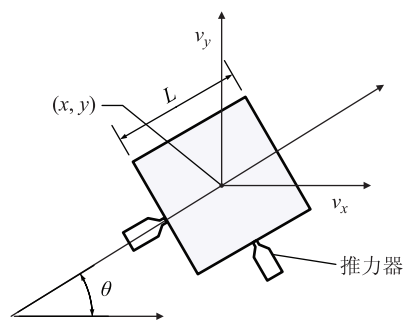


图 2 带有双推力器的平板运动模型

Fig.2 Motion model of a planar body with two thrusters

带有双推力器的平板模型^[13]如图 2 所示, 其状态变量为 $(x, y, \theta, v_x, v_y, \omega)$, 其中 (x, y) 为平板的

中心坐标, θ 表示平板的前进方向, v_x 、 v_y 分别是沿 x 轴、 y 轴方向的平移速度, ω 是平板的旋转速度. 平板的控制输入为两个推力器供给的加速度, 分别为 u_0 和 u_1 (单位为 m/s^2). 平板的长宽均为 L . 运动约束方程为

$$\begin{pmatrix} dx/dt \\ dy/dt \\ d\theta/dt \\ dv_x/dt \\ dv_y/dt \\ d\omega/dt \end{pmatrix} = \begin{pmatrix} v_x \\ v_y \\ \omega \\ 0 \\ g \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ \cos \theta \\ \sin \theta \\ 0 \end{pmatrix} u_0 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\sin \theta \\ \cos \theta \end{pmatrix} u_1 \quad (10)$$

其中 g 为纵坐标方向上的重力偏移 (单位为 m), 控制输入 u_0 和 u_1 需要满足一定的限制条件: $|u_0| \leq u_{0,\max}$, $|u_1| \leq u_{1,\max}$, 转动角速度满足约束 $|\omega| \leq \omega_{\max}$.

在仿真过程中设定: $u_{0,\max} = 0.5 \text{ m/s}^2$, $u_{1,\max} = 0.5 \text{ m/s}^2$, $\omega_{\max} = 100^\circ/\text{s}$, $v_{\max} = 3 \text{ m/s}$, $L = 0.5 \text{ m}$, $\Delta t = 0.3 \text{ s}$.

4.3 单轮车模型

单轮车模型^[13]如图3所示, 其状态变量为 $(x, y, \theta, v, \omega)$, (x, y) 为车轮圆心坐标, θ 表示单轮车的前进方向, r 为车轮半径, v 表示平移速度, ω 表示旋转角速度. 输入控制变量分别为平移加速度 u_0 (单位为 m/s^2) 与旋转角加速度 u_1 (单位为 $^\circ/\text{s}^2$). 运动约束方程为

$$\begin{pmatrix} dx/dt \\ dy/dt \\ d\theta/dt \\ dv/dt \\ d\omega/dt \end{pmatrix} = \begin{pmatrix} v \cos \theta \\ v \sin \theta \\ \omega \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} u_0 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_1 \quad (11)$$

在仿真过程中设定: $u_{0,\max} = 1 \text{ m/s}^2$, $u_{1,\max} = 25^\circ/\text{s}^2$, $\omega_{\max} = 100^\circ/\text{s}$, $v_{\max} = 3 \text{ m/s}$, $\Delta t = 0.5 \text{ s}$.

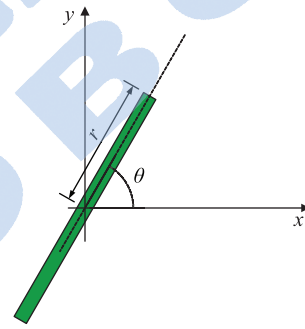


图3 从上方看下去的单轮车模型
Fig.3 A unicycle from the top sight

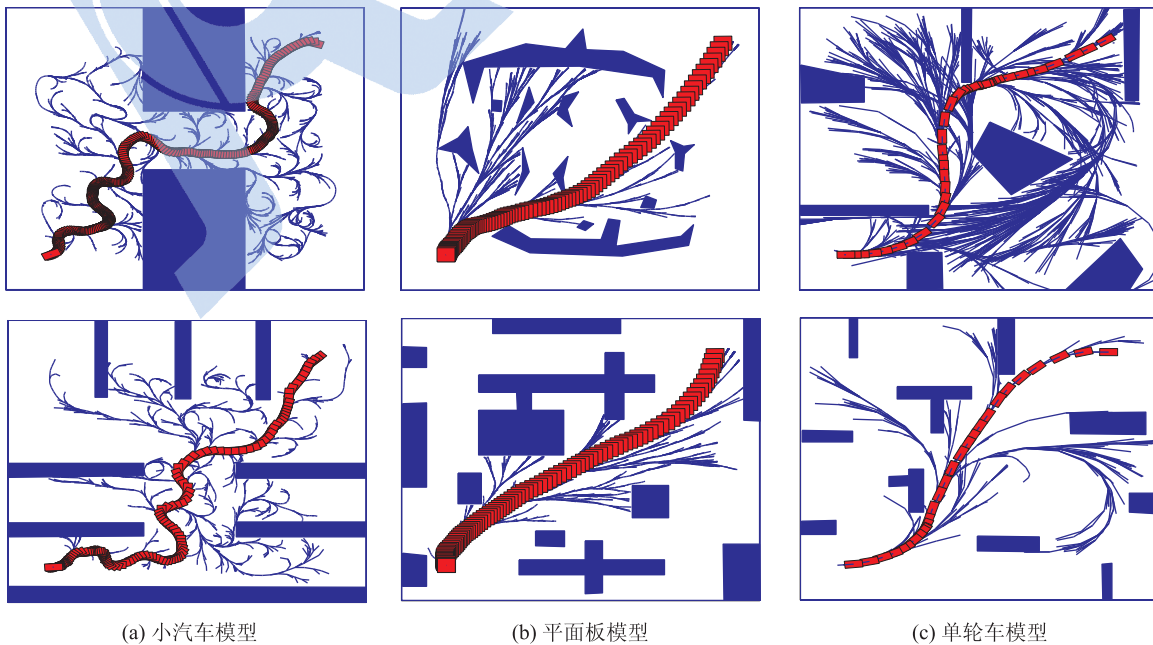


图4 不同障碍物环境下改进的 RRT 算法仿真结果图

Fig.4 Simulation results of the improved RRT for different environment maps

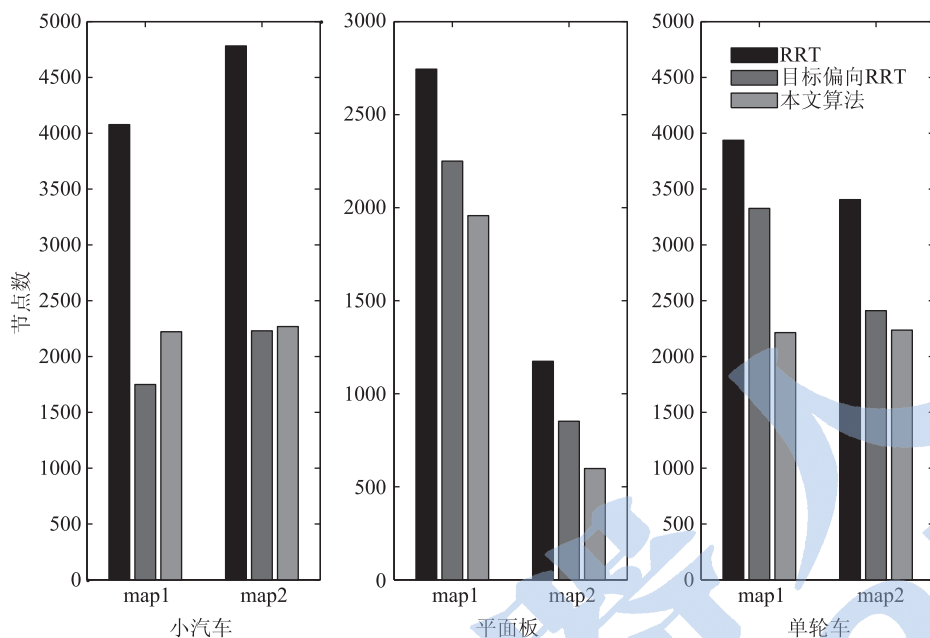


图 5 3 种算法节点对比

Fig.5 Vertex number of the three algorithms

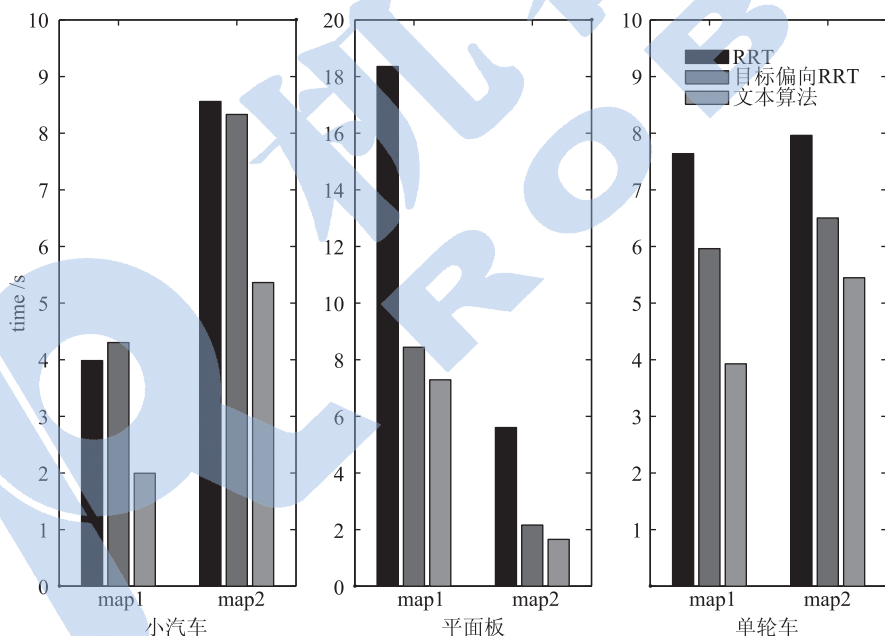


图 6 3 种算法所需时间对比

Fig.6 Time comparison of the three algorithms

4.4 对比实验及分析

由于本文提出的是基于 RRT 算法的改进算法, 目标偏向思想近年来经常被研究者运用, 因此本文选择基本 RRT、目标偏向 RRT 作为对比算法. 为科学评测算法的有效性, 本文采用 6 种障碍物复杂度不同的环境进行验证, 如图 4 所示. 环境为 $10\text{m} \times 10\text{m}$ 的矩形区域, 障碍物随机设置, 大小任意, 起始点坐标为 (1.25, 1.25), 目标点坐标为 (8.75, 8.75). 考虑 RRT 算法本身的随机性, 为客观评价算法性能,

每种环境均进行 20 次运动规划求取时间平均值, 随机树扩展节点数由于直接影响算法速度及系统所需存储空间, 因此也作为评价参数进行对比. 实验在 Intel Core(TM)2、3 GHz、2G 内存的计算机上进行, 编程环境为 Matlab 7.8.0.

不同环境下以小汽车等作为模型的运动规划结果如图 4 所示, 限于篇幅此处只给出每种模型在两种环境下的仿真结果. 验证结果表明本文提出的改进算法对于受非完整约束限制的机器人运动规划

具有可行性，在环境中存在可行路径的情况下该算法都能获得有效解，搜索成功次数高于原 RRT 算法与目标偏向 RRT 算法。三种算法在获得有效路径时扩展树平均节点个数对比如图 5 所示，相比于原 RRT 算法，由于引入目标偏向思想，另外两种算法节点数大大减少，分别为原 RRT 算法节点数的 63.7%、57.1%，在实际应用中能有效降低系统对存储空间的要求。由图 6 的 3 种算法所需时间对比可以看出，目标偏向 RRT 算法相对于原 RRT 算法并不占优势，而本文提出的算法由于加入距离参数选择机制，选择计算复杂度低的对角线距离参数提高了求解速度，算法平均耗时仅为原 RRT 算法耗时的 62.2%。

由于 bi-RRT 算法在目前路径规划研究中很常见，本文也与 bi-RRT 算法进行了对比。以小汽车模型为例，选取如图 7 所示的两种环境进行分析。两种算法的运行时间如表 1 所示。由图中可以看出，对于“T”型环境，由于搜索树易陷入局部最小，两棵树同时搜索所需平均时间要高于本文提出的算法；对于另一种环境，两种算法所需平均时间相近。但 bi-RRT 算法应用于非完整约束条件下的机器人运动规划时，在两棵树相遇处存在不连续现象，需借助其他算法进行处理以保证在连接处符合约束条件（本文在仿真中忽略），增加了算法复杂度。

表 1 算法运行结果对比

Tab.1 Results comparison

平均时间 /s	环境 1	环境 2
bi-RRT	6.2717	8.3452
改进 RRT	3.3779	7.3125

5 结论 (Conclusion)

非完整约束普遍存在于轮式机器人、汽车、飞行器和船舶等系统中，运动规划问题又是移动机器人实现自主导航等应用时必须解决的问题。以往的路径规划算法大多只考虑完整约束，假设机器人只要避开障碍物就可以在位形空间内任意运动，这样规划出的路径不一定对应工作空间内的一条可行路径。本文提出一种改进的 RRT 算法，将非完整性约束条件与 RRT 搜索算法相结合，实验表明该算法能有效解决存在非完整约束的机器人运动规划问题。

本文提出的改进 RRT 算法适用于全局环境已知的运动规划问题，可考虑将该算法与基于滚动窗口的路径规划算法结合，使其在未知环境下的运动规划问题中仍有所应用。另外，为使非完整约束下的机器人运动规划问题满足实际应用中对实时性、完备性的需求，多棵随机树同时搜索、综合最短路算法与采样算法优点的混合算法等都是值得研究的方向。

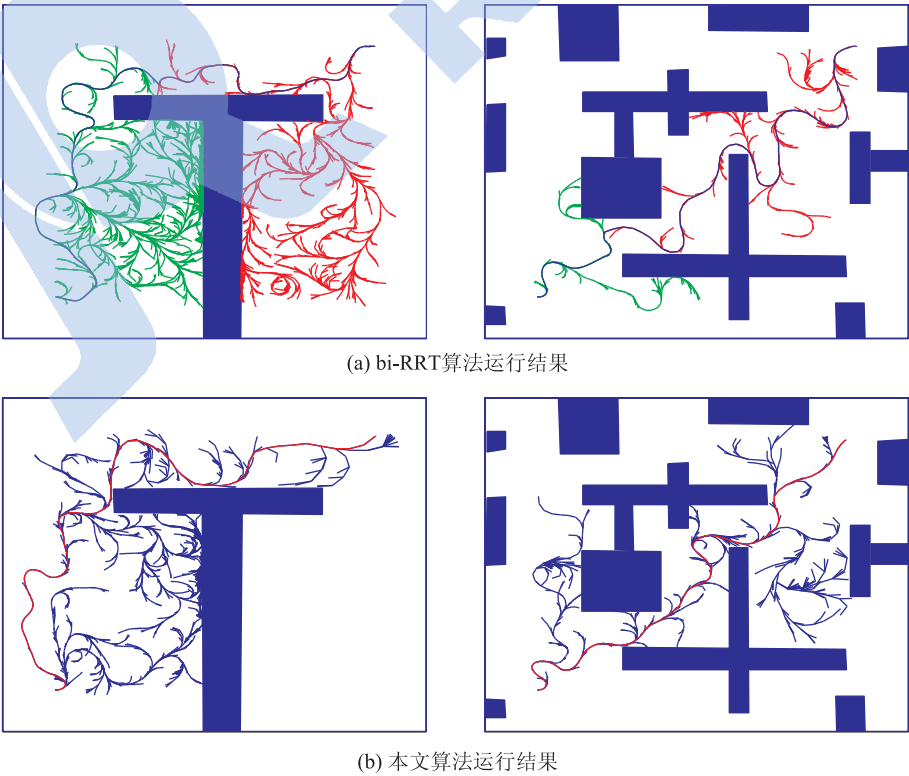


图 7 bi-RRT 算法与改进 RRT 算法仿真结果图

Fig.7 Results of the simulation with bi-RRT and the improved RRT

参考文献 (References)

- [1] Arkin R C. Behavior-based robotics[M]. 1st ed. Cambridge, USA: MIT Press, 1998: 1-3.
- [2] Lozano-Perez T. Spatial planning: A configuration space approach[J]. IEEE Transactions on Computers, 1983, 32(2): 108-120.
- [3] LaValle S M. Rapidly-exploring random trees: A new tool for path planning[R]. Iowa, USA: Computer Science Department, Iowa State University, 1998.
- [4] LaValle S M, Kuffner J J. Randomized kinodynamic planning[J]. International Journal of Robotics Research, 2001, 20(5): 378-400.
- [5] 康亮, 赵春霞, 郭剑辉. 基于模糊滚动 RRT 算法的移动机器人路径规划 [J]. 南京理工大学学报: 自然科学版, 2010, 34(5): 642-648.
- Kang L, Zhao C X, Guo J H. Path planning based on fuzzy rolling rapidly-exploring random tree for mobile robot[J]. Journal of Nanjing University of Science and Technology: Natural Science, 2010, 34(5): 642-648.
- [6] Kuffner J J, LaValle S M. RRT-connect: An efficient approach to single-query path planning[C]//IEEE International Conference on Robotics and Automation. Piscataway, NJ, USA: IEEE, 2000: 995-1001.
- [7] Rodriguez S, Tang X Y, Lien J M. An obstacle-based rapidly-exploring random tree[C]//IEEE International Conference on Robotics and Automation. Piscataway, NJ, USA: IEEE, 2006: 895-900.
- [8] Szadeczky-Kardoss E, Kiss B. Extension of the rapidly exploring random tree algorithm with key configurations for nonholonomic motion planning[C]//IEEE International Conference on Mechatronics. Piscataway, NJ, USA: IEEE, 2006: 363-368.
- [9] Ferguson D, Stentz A. Anytime RRTs[C]//Proceedings of the International Conference on Intelligent Robots and Systems. Piscataway, NJ, USA: IEEE, 2006: 5369-5375.
- [10] de Smith J. Distance and path: The development, interpretation and application of distance measurement in mapping and modeling[D]. London, UK: University College of London, 2003.
- [11] 刘磊, 向平, 王永骥, 等. 非完整约束下的轮式移动机器人轨迹跟踪 [J]. 清华大学学报: 自然科学版, 2007, 47(S2): 1884-1889.
- Liu L, Xiang P, Wang Y J, et al. Trajectory tracking of a non-holonomic wheeled mobile robot[J]. Journal of Tsinghua University: Science and Technology, 2007, 47(S2): 1884-1889.
- [12] LaValle S M. Planning algorithms[M]. Cambridge, UK: Cambridge University Press, 2006: 722-724.
- [13] Choset H, Lynch K M, Hutchinson S, et al. Principles of robot motion: Theory, algorithms, and implementations[M]. Cambridge, USA: MIT Press, 2005: 406-407.

作者简介:

徐娜 (1986-), 女, 硕士生. 研究领域: 机器人运动规划, 多机器人协同, SLAM.

陈雄 (1964-), 男, 博士, 副教授. 研究领域: 控制理论与系统, 移动机器人控制与运动规划, 传感器网络等.

孔庆生 (1961-), 男, 硕士, 副教授. 研究领域: 嵌入式系统软硬件设计, 工业控制与自动化, 计算机控制与管理系统, 智能仪器仪表.

(上接第 665 页)

作者简介:

卢翔 (1987-), 男, 博士生. 研究领域: 图像处理, 视觉伺服, 网络机器人.

刘景泰 (1964-), 男, 教授, 博士生导师. 研究领域: 机器人技术, 计算机应用技术, 信息化工程.