

一种基于改进的遗传算法的块匹配运动估计方法

龚涛 丁润涛

天津大学 电子信息工程学院, 天津 300072

摘要 块匹配方法 (*Block Matching Algorithm*, 简称*BMA*) 是目前广泛使用的运动估计方法, 但该方法的最大缺点是容易陷于局部最优, 这主要是由搜索模式决定的。而遗传算法 (*Genetic Algorithm*, 简称*GA*)^[1] 是一种具有广泛适应性的全局最优的搜索算法。将块匹配方法的局域性搜索与遗传算法的全局性搜索结合起来, 本文提出了一种基于改进的遗传算法的块匹配运动估计方法。实验证明, 该方法的平均绝对误差 (*MAE*) 接近全搜索 (*FSS*), 优于三步法 (*TSS*), 而运算量相对较低, 接近三步法。

关键词 遗传算法 (*GA*), 最小均方误差 (*MSE*), 平均绝对误差 (*MAE*), 单一平面假设 (*UESA*)

A Modified Genetic Algorithm based Block Matching Motion Estimation Method

GONG Tao, DING Run-tao

School of Electronic Information Engineering, Tianjin University, Tianjin 300072

Abstract: The Block Matching Algorithm (BMA) is currently widely used in Motion Estimation, but it is suboptimum and susceptible to be trapped into local optimum due to its specific searching pattern. While, Genetic Algorithm (GA) is a global optimum searching method used in many fields which require global optimum from large data. This paper combines BMA definite local searching with GA elective global searching and proposes a block matching algorithm based on a modified GA. The simulations show that the Mean Absolute Error (MAE) performance of this new algorithm is similar to that of FSS, better than TSS, while the computation complexity of it is lower than that of FSS and similar to TSS.

Key words: Genetic Algorithm (GA), MSE, MAE, UESA.

1 引言

运动估计是消除图像帧间冗余的有效方法。在 H.263, MPEG-4 的图象序列中, 对象在相邻的几帧中运动不大, 块匹配方法简单有效得到广泛应用。块匹配方法的实现效果取决于 3 个因素: 搜索范围, 匹配准则和搜索模式。搜索范围可根据运动的剧烈程度进行调整以获得精确的匹配; 匹配准则是决定搜索精度的重要因素, 目前主要使用的有两个: 最小均方误差 (*MSE*) 和平均绝对误差 (*MAE*), 其定义如下:

$$MAE(x, y) = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |(F_n(i, j) - F_{n-1}(i+x, j+y))| \quad \dots (1)$$

$$MSE(x, y) = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (F_n(i, j) - F_{n-1}(i+x, j+y))^2$$

... (2)

其中, $-W \leq x, y \leq W$, F_n 和 F_{n-1} 分别为当前块和前一帧中的对应块, 大小均为 $N \times N$, MPEG-4 标准中, N 取 16。 W 为运动矢量在水平、垂直方向上最大位移, 搜索窗大小为 $(2W+1)^2$ 。为减少运算量和便于硬件实现, 匹配准则多采用 *MAE*; 不同的块匹配方法有不同的搜索模式, 如菱形法^[2], 六边形法^[3]等, 它们都是基于单一平面假设 (*UESA*)^[4] (认为当远离全局最优匹配点时, 匹配误差随距离呈线性增加) 而提出的。这种特定的搜索模式导致了目前的块匹配方法很容易陷入局部最优而不一定是全局最优, 尤其在较复杂的运动情况下。

由 John Holland 在 70 年代提出的遗传算法 (*GA*) 是一种通过编码产生染色体 (*Chromosome*), 通过计算适应性方程 (*fitness function*)、选择 (*Selection*)、

交叉 (Crossover) 和变异 (Mutation) 等操作进行迭代搜索以获得最优解。它抛开具体搜索空间、数据连续性等限制, 在全局范围内搜索最优值。

本文结合块匹配方法的局部特性和遗传算法的全局特性, 提出一种基于改进的遗传算法的块匹配方法, 并给出实验结果。全文结构如下: 第 1 部分, 介绍本文方法; 第 2 部分, 给出实验结果; 第 3 部分, 讨论并提出改进措施作为未来研究方向。

2 基于改进遗传算法的块匹配方法

本文块匹配方法中, 宏块大小为 16×16 , 搜索窗为 $(2 \times 15 + 1)^2$, 即 $W=15$ 。匹配准则采用 MAE , 匹配条件: $MAE \leq \varepsilon$, ε 是预先指定的参数, 由要求的精度和先验经验决定。下面论述该方法具体步骤。

1, 染色体编码。 BMA 搜索匹配点, 对搜索窗内的点 (x, y) 编码产生染色体, 单个坐标编码长度为:

$\lceil \log_2 W + 1 \rceil = 5$, 总的染色体长度为 $L=10$ 。这里对搜索窗内的点相对于搜索窗中心的位移进行编码, 而不是根据实际位置进行编码。编码方式采用格雷码, 此种码相邻的码字只有 1 位不同, 在变异操作中容易移到相邻位置。比如, 点 $(-2, 3)$, 表示该点位于搜索窗中心点的左侧 2 单位、下方 3 单位处, 用格雷码编码 (符号位规定 1 为负, 0 为正) 形成的染色体为 $(10011\ 00010)$ 。这样找到的匹配点坐标可直接作为运动矢量。综上, 染色体为: 其中 $m=4$,

$$C_i = \{x_i, y_i\} = \{x_{i,m}, x_{i,m-1}, \dots, x_{i,0}, y_{i,m}, y_{i,m-1}, \dots, y_{i,0}\}$$

$$x_{i,j} = Gray_j^i(x_i - x_0), \quad y_{i,j} = Gray_j^i(y_i - y_0)$$

(x_0, y_0) 为搜索窗中心点坐标, $x_{i,4}, y_{i,4}$ 是符号位。

2, 初始染色体个数和位置。传统的遗传算法中, 初始个体是随机产生的, 其个数和位置决定着能否快速找到最优解。个数过少、分布过于集中, 迭代可能过早收敛; 个数过大, 运算量较大; 分布过稀, 迭代次数较多。在 BMA 具体问题中, 应根据运动序列的具体情况指定初始点。首先, 运动矢量具有中心偏移特性。其次, 相邻宏块运动相似, 周围已经估计过的宏

块的运动矢量可作为本块的预定运动矢量, 如图 1 所示。根据以上两个特性, 初始染色体个数为 9 个, 位置指定方法如下: a , 对于图象边缘的宏块, 没有参考宏块, 初始点为搜索窗口中心的 9 个点, 如图 2 (a) 所示; b , 对于内部宏块, 根据周围匹配过的宏块预先设定运动矢量 W_i , 初始点为该运动矢量周围的 9 个点, 如图 2 (b) 所示^[7]。

3, 适应性函数和选择操作。遗传算法最优值为适应度大的点, 而 BMA 中最优点是使 MAE 值最小的点, 两者是统一的。适应性函数定为: $F(i) = -MAE$, MAE 越小的点, 其适应度越高 ($Guanghua\ Qiu$ 在文献 [5] 中使用 $F(i) = 1/MAE$, 本文中省略了该文献中的倒数运算, 降低了计算复杂度和存储空间)。同时, 直接观察 MAE 的值比较直观。一代群体中的个体按适应度大小排序, 进行选择。不采用传统的轮盘赌 (Rotational Wheel) 方法^[6], 本方法直接选择适应度大的前 $\lfloor N/2 \rfloor$ 个个体进入下一代, 这种选择考虑了图象数据的空间连续性, 省去了多余的计算 (如平均适应度等)。实践证明, 在不大的搜索窗 (256 点左右) 中, 这种方法和轮盘赌方法的效果相似。剩下的个体进行其它操作后再进入下一代。

4, 交叉和变异。这是遗传算法中产生新个体的主要手段, 操作如图 3 所示。对于二进制编码, 交叉和变异都是使染色体某一个或几个基因座上的值取反或不变, 依次对每位进行变异在概率上与交叉的效果相同。在 BMA 问题中, 染色体是两个坐标的编码拼接而成, 它们之间没有必然联系, 截取几位进行交叉的操作没有实际意义, 而变异是针对位进行的, 对 x 坐标和对 y 坐标变异是独立的。同时, 由于最优匹配点位于适应度高的点的周围, 对位变异有可能快速找到最优点, 而交叉很可能造成在最优点周围摆动。最后, 变异操作简单, 利于软硬件实现。所以, 本方法省略交叉操作只进行变异操作, 并指定一个较大的变异概率 (P_m) (文献 [5]~[7] 中在有交叉环节的情况下通常令 $P_m=0.05$, 本文省略交叉, 经过实验令 $P_m=0.25$, 证明

变异效果较好), 来增加新个体产生的机会, 尽快定位最优匹配点。具体实施步骤如下: 针对染色体的每个基因座, 随机产生 $(0, 1)$ 间的数, 如果大于 P_m , 则将该基因座上的值取反; 否则, 该值保持不变。符号位进行同样处理, 使新个体产生的范围增大, 以免局限于某个局部最优值。变异操作后, 剩下的 $(N - \lfloor N/2 \rfloor)$ 个染色体和直接选择的染色体一起组成下一代群体。变异是本方法产生新个体的唯一措施。另外, 在新一代个体中, 只需计算经过变异产生的新个体的适应度, 节省了部分运算。

5, 迭代结束条件和追加搜索。为减少迭代次数, 同时充分利用 *BMA* 的局部搜索特性。迭代结束条件定为: a , 迭代次数达到最大值: $I = \lfloor \log_2 2L \rfloor = 4$, 这是根据染色体长度得出的, 保证多数情况下搜索收敛的一个先验常数。如果迭代次数达到, 则取这代中的最优解作为最终结果 (实际上, 大部分搜索都已经提前收敛, 因为有结束条件 b 的支持)。这是全局限制; b , 某代中最有个体满足匹配条件, 这是局部性限制。结合 *BMA* 的局部搜索特性引入追加搜索: 搜索每一代中的最优点 (适应度最高) 周围的 8 个点, 计算 *MAE*, 如果有满足匹配条件的点, 则搜索结束; 否则, 进行新一轮迭代。追加搜索是根据 *BMA* 问题中, 考虑了图象数据的空间联系性而加入的, 可以减少遗传操作的运算量, 同时有利于快速定位最优匹配点。

综上, 本文方法的流程图如图 4 所示。对遗传算法主要进行如下改进: 简化了选择操作, 省去了交叉操作, 初始个体位置预先设定。下面给出使用本算法的实验结果。

3 实验结果

实验序列为 *Football* (较剧烈运动) 序列、*Miss* 序列 (微小运动) 和 *Dr.Aki* 序列 (斜向平动为主), 搜索窗为 (± 7) 。为比较, 给出 *FSS* 和 *TSS* 的运算结果如表 1 所示。本文方法的计算量主要集中在 *MAE* 的计算量 (设为 $M1$, 与平均搜索点数成正比) 和遗传算法中

的排序时进行比较 (设为 $M2$, 与群体中的个体数目有关)、变异 (设为 $M3$, 与群体中个体数目和个体染色体编码长度有关) 操作。本文中, 群体中个体数目为 9, 每代中有 5 个个体需要进行变异操作, 染色体编码长度为 10, 所以运算量可用式(3)计算:

$$C = n \times M1 + t \times M2 + 10 \times t \times 5 \times M3 \quad \dots(3)$$

其中, n 表示平均搜索点数, t 表示平均遗传算法的迭代次数。第一部分的运算量占全部运算量的很大部分。

不同的搜索方法 (如全搜索、三步法等) 对每个搜索点在 *MAE* 的计算量上 (假设搜索窗相同, 宏块大小相同) 是类似的。峰值信噪比: $P_{snr} = 20 \times \log_{10}(255/\sqrt{MSE})$, 本文采用 *MAE*, 该参数可简化为 $P_{snr} = 20 \times \log_{10}(255/MAE)$, 它可以衡量算法的精度。从表 1 中可见, 本文方法的精度高于 *FSS*、*TSS*; 而平均搜索点数接近 *TSS*, 少于 *FSS*, 尽管运算时间大于 *TSS* (因为平均搜索点数较多, 加之变异等辅助操作), 但本方法对于不同的运动情况, 运算时间相差不多。也就是说, 运算时间不随具体问题而有较大的变化。这是因为遗传算法在全局进行搜索, 对染色体操作, 突破了具体问题 (如最优点的位置等) 的限制, 所以本文方法具有稳定性和普适性。

4 讨论和结论

本文将遗传算法引入运动估计, 结合遗传算法的全局搜索和运动估计的局部搜索进行 *BMA*, 对遗传算法进行了改进。通过对实验的分析, 可以看出本方法的一些特点: 逐点变异产生新个体的效果与交叉、变异组合作用的效果近似, 而计算上简单快速; 文献[5]、[6]中的方法, 初始染色体个数为 16 或 18 个, 本文方法根据预设的运动矢量指定初始点并且引入追加搜索, 所以 9 个初始染色体就可以达到预期效果, 染色体个数减少, 分布对称, 相应的遗传操作的运算量也随之降低。初始个体数目和位置决定了运算的收敛速度。同时, 追加搜索使大部分估计 (对 *Dr.Aki* 序列, 约 75% 以上) 都在最大迭代数之前结束, 部分节省了运

算量。最后，针对新版的MPEG-4标准^[8]中引入的半像素估计，对本方法稍加改进就可以直接应用。即在本方法追加搜索时，用插值公式求出半像素点（分布如

$$MAE(x,y)_{1/2,a,b} = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |(F_n(i,j) - (F_{n-1}(i+x,j+y) + F_{n-1}(i+a+x,j+b+y))/2)| \quad (4)$$

其中，a, b 分别取 (-1, 0, +1)。这样，先找到整像素最优匹配点，再在追加搜索时利用插值公式计算该最优点周围的半像素点找到最优匹配点。具体实现本文从略。

为进一步降低运算量，本方法可以进行如下改进。首先，基于MPEG-4、H.263标准的图象序列，它们的精度要求不高，宏块内大部分点的运动都比较接近，只估计其中的几点，就可以得到整个宏块的运动矢量。根据这一点，可以先对图象进行预处理，如通过下采样^[9]、小波分解等，得到低分辨率图象，再应用遗传算法搜索。总搜索空间的点数降低，使得遗传算法的迭代次数相应降低，运算量也随之降低。

其次，并行遗传算法^[8]可以降低运算量，其基本思想是：将初始搜索空间分成几个子空间，在子空间用较小的群体进行迭代，一定次数后，比较各子空间中群体的当前最优解以获得全局最优解。由于子空间中群体规模较少，所以在同样的迭代次数下，可以更

快收敛到最优解，而且全局最优解就在子空间群体的最优解中。基于并行遗传算法，本文方法可进行如下改进：先将搜索窗划分成4个子窗，图6给出了两种划分方式。增加初始点个数为16，每个子窗中4个初始点，随机产生。在子窗中进行遗传迭代，每进行一次迭代，检查原搜索窗中心点和子窗中当前最优解的MAE是否满足匹配条件，若有点满足，算法结束；否则继续迭代直到达到次数限制。这里没有预设运动矢量的限制和追加搜索（搜索空间较小），算法更灵活。同时编码方式也可以更灵活，比如图6左面的子窗划分方法，编码时可以避开正负混合编码。对于运算量，在子窗内进行迭代，4个子窗1次迭代的计算量和母窗1次迭代差不多，只是增加了几个初始点的MAE和变异计算；而在点数较少的子窗内进行迭代，可以很快达到最优点，所以总的迭代次数比原来少很多（大约减少1/3）。所以，运算量有显著下降。

宏块1 运动矢量: W_1	宏块2 运动矢量: W_2
宏块3 运动矢量: W_3	待估计块 运动矢量: W_i

$$W_i = (W_1 + W_2 + W_3) / 3$$

图1，围宏块与待测块分布

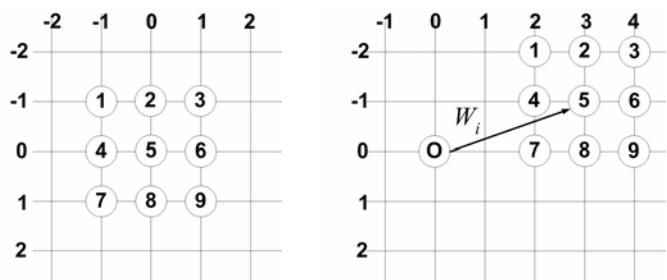


图2，初始点分布示意图（O点为搜索窗中心， $W_i(3,-1)$ 为预设的运动矢量）

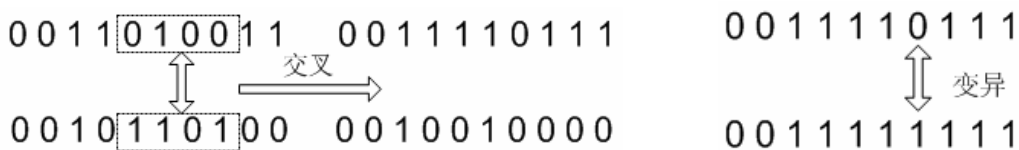


图3，双交叉点交叉和单点变异操作示意图

表 1, 本文方法和 FSS、TSS 性能比较

性能 \ 算法种类	全搜索 (FSS)			三步法 (TSS)			本文方法		
	Football	Dr. Aki	Miss	Football	Dr. Aki	Miss	Football	Dr. Aki	Miss
平均搜索点数	255	255	255	18.43	13.88	14.75	19.36	18.92	18.32
运算时间 (秒)	758.57	760.29	758.94	97.9	91.65	92.55	115.56	113.05	112.36
峰值信噪比 (PSNR) (db)	25.52	25.98	35.44	25.33	26.75	35.07	28.50	27.50	37.35

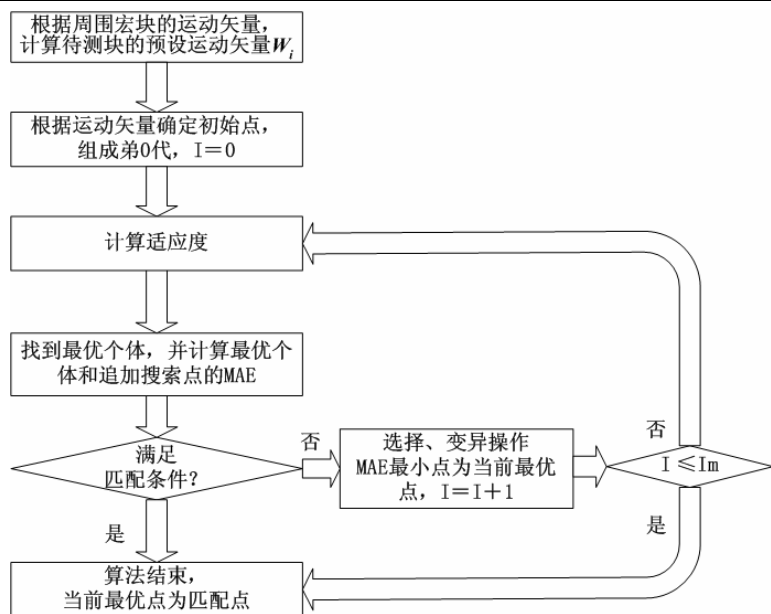


图 4, 本文算法流程图

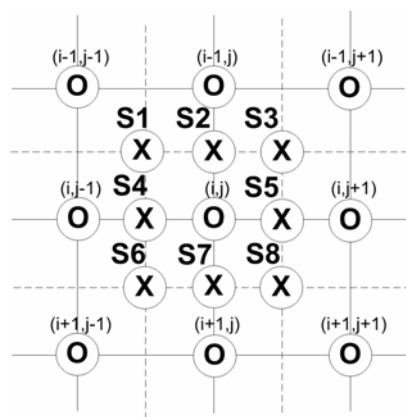


图 5, 半像素点分布, O 为当前整像素最优点, a 为整像素点, b 为半像素点

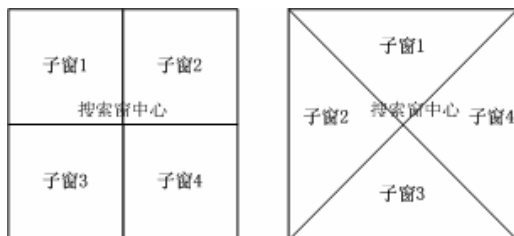


图 6, 子窗划分示意图

参考文献

[1] J. H. Holland, *Adaptation in Natural and Artificial Systems* [M], 1st ed., Michigan University Press, 1975; 2nd ed., Cambridge, MA: MIT Press 1992.

[2] 洪波, 余松煜. 基于对象的菱形搜索运动估计方法[J]. 数据采集与处理, 2001, 16(1): 110~116.

[3] Zhu Ce, Lin Xiao. A novel hexagon-based search algorithm for fast block motion estimation [A]. In: IEEE ICA SSP'01 [C], Singapore, 2001: 1593~1596.

[4] 洪波, 庄建敏, 余松煜. 基于时空相关性的自适应运动估计方法[J]. 计算机工程, 2000, 8: 50~53.

[5] Guanghua Qiu, Chao-hua Hou, A new fast Algorithm for the Block Motion Vector [A], In Proc. of ICSP'96, 1996, Beijing: 1233~1236.

[6] Xu Yuelei, Bi Duyan, Mao Baixin, A Genetic Algorithm for Motion Estimation [A], In Proc. of ICSP2000 [C], 2000, Beijing: 1058~1062.

[7] Man F. So, Angus Wu, Four-Step Genetic Search for Block Motion Estimation [J], IEEE Trans. of SP, 1998, 2(2) : 1393~1397.

[8] MPEG. ISO/IEC JTC1/SC29/WG11 Document N3156, Overview of MPEG-4 Standard [EB], <http://www.mpeg.org/>, Dec. 1999.

[9] 程世龙, 戴卫桓, 程红煌 等, 基于进化规则的运动估计算法 [J]. 通信学报, 2001, 6(22) : 113~116