

多分辨逐步消除快速匹配算法

马社祥^{1,2}, 刘铁根¹

(1. 天津大学 精密仪器与光电子工程学院, 天津 300072;

2. 天津理工大学 光电信息与电子工程系, 天津 300191)

(masx@eyou.com)

摘 要: 为了提高块匹配运动估计的速度和精度, 提出了多分辨逐步消除快速匹配算法。在块匹配运动估计中, 像素块通过抽取降样而构成四种分辨率, 对某匹配点而言, 若其在较低分辨率上不满足为最优的必要条件, 即对应于一不等式成立, 说明该匹配点与当前最优点的像素信息差别较大, 因此, 将其消除, 而不必进行复杂的匹配运算; 否则, 再在较高分辨率上进行相同的操作过程。如此逐步精细求优过程, 使得只需用少量的计算就可以消除相当一部分的匹配点。同时, 结合 NTSS 快速算法进行多分辨逐步消除。实验结果显示, 本文方法能有效减少运算量。

关键词: 多分辨; 逐步消除; 运动估计

中图分类号: TP311 **文献标识码:** A

Multiresolution successive elimination block matching

MA She-xiang^{1,2}, LIU Tie-gen¹

(1. College of Precision Instrument and Optoelectronics Engineering, Tianjin University, Tianjin 300072, China;

2. Department of Optoelectronics Information and Electronics Engineering, Tianjin University of Technology, Tianjin 300191, China)

Abstract: In order to increase the computational speed of the motion estimation and to improve the image quality of the motion compensation in video coding, a multiresolution successive elimination block-matching algorithm was introduced. In motion estimation, four kinds of resolution are obtained by down sampling of every pixel block. For one match point, if the optimal necessary condition is not satisfied in the lower resolution, then the point will be eliminated by the spatial correlation of pixels. Otherwise, the process is operated in the higher resolution again. By the way, a lot of match points will be eliminated with a few amount of computation. Meanwhile, the idea of NTSS was also used to the multiresolution successive elimination algorithm. Experimental results show that the scheme is effective.

Key words: multiresolution; successive elimination; motion estimation

0 引言

块匹配运动估计和运动补偿技术以其简单有效而被广泛应用于视频压缩编码中, 成为消除视频帧间相关性的最主要的方法之一。在块匹配运动估计中, 主要采用全搜索法 (FS) 进行搜索匹配, 这种方法虽然简单可靠, 但时间复杂度太大。因此, 各种快速算法被提出, 文献[1]中提出了二维对数法 (Two-Dimensional Logarithmic Search, TDL), 其搜索过程是跟踪最小绝对均方差 (MAD), 每次检查一个十字形分布的 5 个像素构成的点群, 算出 5 个 MAD 值, 看哪一个 MAD 最小。若最小 MAD 出现在十字形点群处于边上的点处, 将该点作为十字形点群的搜索中心, 步长不变, 即把十字向最小 MAD 方向移动。反之, 若十字形点群的中心点处的 MAD 最小, 则下一步的搜索不改变中心点而把十字形缩小。依此类推, 直到找到 MAD 的极小点。若在上述搜索过程中发现新的十字形的中心处于搜索区的边缘, 则步长也减小。二维对数法的搜索区中有一些点是其第一步永远也搜索不到的, 而且搜索的随意性较大。另一个著名的快速搜索算法是三步搜索

法 (Three-Step Search, TSS)^[2], 和二维对数法相类似, 三步法也是一个由粗到细的逐步搜索最优的算法。刚开始时以 $(v_x, v_y) = (0, 0)$ 点为中心, 最大搜索长度的一半为步长, 检测中心点以及周围 8 个邻点的 MAD 值, 找到最小点。下一步以此为中心, 步长减半, 检测中心点以及周围 8 个邻点的 MAD 值最小的点。依此类推, 直到搜索精度达到要求值为止。除此之外, 还有改进的三步搜索算法 (NTSS)^[3]、正交搜索法^[4]、交叉法^[5]、一维全搜索法^[6]、四步搜索法^[7]和两步搜索法^[8]等。

快速匹配算法主要通过减少匹配点数来减少运算量, 可能会使寻优过程陷入局部最优。近年来出现了一种新的快速运动估计匹配算法, 称为逐步消除算法 (SEA)^[9~12], SEA 可以求得全局最优匹配点, 同时提高了运动估计速度, 尤其对较简单的视频序列, 象可视电话和会议电视等, 其效果更为突出。对较简单的视频序列来说, 其时间和空间相关性更强, 因此在减少匹配点数和降样下的逐步消除法, 可使得在质量下降不多的情况下能更进一步提高匹配速度。基于此, 本文提出了多分辨逐步消除快速匹配算法 (MSEA)。

收稿日期: 2005-08-13; 修订日期: 2005-10-27

基金项目: 天津市高等学校科技发展基金资助项目 (20041322); 中国博士后科学基金资助项目 (2004036358)

作者简介: 马社祥 (1962-), 男, 甘肃宁县人, 教授, 博士, 主要研究方向: 图像处理; 刘铁根 (1955-), 男, 教授, 博士生导师, 主要研究方向: 光电、光纤检测技术。

1 逐步消除算法

1.1 SEA 原理

SEA^[9] 的主要思想是逐步消除那些不可能是最优点的匹配点。假设匹配准则是绝对差和(SAD)最小准则,搜索范围记为 W 。从 FS 算法来分析 SEA 的基本原理。FS 的搜索是通过如下步骤来实现:

第一步:初始化,假设 (m, n) 是当前的最优匹配点,即 $SAD(m, n)$ 最小;

第二步:逐步搜索,对任一点 $(x, y) \in W$

$$1) \text{ 计算 } SAD(x, y) = \sum_{j=1}^N \sum_{i=1}^N |I_t(i, j) - I_{t-1}(i+x, j+y)| \quad (1)$$

$$2) \text{ 若 } SAD(x, y) < SAD(m, n) \quad (2)$$

则 $(m, n) = (x, y)$ 。

$$\text{由于 } ||I_t(i, j)| - |I_{t-1}(i+x, j+y)|| \leq |I_t(i, j) - I_{t-1}(i+x, j+y)| \quad (3)$$

式(3)等价于下面两个不等式:

$$|I_t(i, j)| - |I_{t-1}(i+x, j+y)| \leq |I_t(i, j) - I_{t-1}(i+x, j+y)| \quad (4)$$

$$|I_{t-1}(i+x, j+y)| - |I_t(i, j)| \leq |I_t(i, j) - I_{t-1}(i+x, j+y)| \quad (5)$$

所以有:

$$\sum_{j=1}^N \sum_{i=1}^N |I_t(i, j)| - \sum_{j=1}^N \sum_{i=1}^N |I_{t-1}(i+x, j+y)| \leq \sum_{j=1}^N \sum_{i=1}^N |I_t(i, j) - I_{t-1}(i+x, j+y)| \quad (6)$$

$$\sum_{j=1}^N \sum_{i=1}^N |I_{t-1}(i+x, j+y)| - \sum_{j=1}^N \sum_{i=1}^N |I_t(i, j)| \leq \sum_{j=1}^N \sum_{i=1}^N |I_t(i, j) - I_{t-1}(i+x, j+y)| \quad (7)$$

记 $R = \sum_{j=1}^N \sum_{i=1}^N |I_t(i, j)|$, 以及 $Norm(x, y) = \sum_{j=1}^N \sum_{i=1}^N |I_{t-1}(i+x, j+y)|$, 则式(6)和(7)可写为:

$$R - Norm(x, y) \leq SAD(x, y) \quad (8)$$

$$Norm(x, y) - R \leq SAD(x, y) \quad (9)$$

即:

$$|Norm(x, y) - R| \leq SAD(x, y) \quad (10)$$

对匹配点 (x, y) , 若式(2)成立, 则由式(10)有:

$$|Norm(x, y) - R| < SAD(m, n) \quad (11)$$

SEA 的思想是只需在使式(11)成立的点上进行匹配求优, 而对不满足(11)的点, 当然也就不满足式(2), 因此将其消除。而计算一个 $Norm(x, y)$, 其运算量是计算一个 $SAD(x, y)$ 的运算量的一半。同时, 对搜索窗内的所有点来说, $Norm(x, y)$ 的计算还可以进一步优化以减少运算量^[9]。

1.2 基于分块的 SEA

若式(8)和(9)的不等式两端越接近, 则由式(10)可以消除更多不必要的点。为此 Jung 等人提出了基于分块的消除法^[10]。

不失一般性, 我们假设将当前要匹配求运动矢量的块 B 分割成不重叠的两个子块 B_1 和 B_2 。分别记:

$$R_1 = \sum_{(i,j) \in B_1} |I_t(i, j)|, R_2 = \sum_{(i,j) \in B_2} |I_t(i, j)|$$

$$Norm_1(x, y) = \sum_{(i,j) \in B_1} |I_{t-1}(i+x, j+y)|$$

$$Norm_2(x, y) = \sum_{(i,j) \in B_2} |I_{t-1}(i+x, j+y)|$$

$$SAD_1(x, y) = \sum_{(i,j) \in B_1} |I_t(i, j) - I_{t-1}(i+x, j+y)|$$

$$SAD_2(x, y) = \sum_{(i,j) \in B_2} |I_t(i, j) - I_{t-1}(i+x, j+y)|$$

$$\text{则 } R_1 + R_2 = R, SAD_1(x, y) + SAD_2(x, y) = SAD(x, y),$$

$$Norm_1(x, y) + Norm_2(x, y) = Norm(x, y)$$

且:

$$|Norm(x, y) - R| =$$

$$|Norm_1(x, y) + Norm_2(x, y) - R_1 - R_2|$$

$$\leq |Norm_1(x, y) - R_1| + |Norm_2(x, y) - R_2|$$

$$\leq SAD_1(x, y) + SAD_2(x, y) = SAD(x, y) \quad (12)$$

由式(2)和(12)有:

$$|Norm_1(x, y) - R_1| + |Norm_2(x, y) - R_2| < SAD(m, n) \quad (13)$$

所以, 对于匹配点 (x, y) , 若不满足式(13), 则将其消除。由不等式(2)和(12)可知, 基于分块的 SEA 可以消除更多的不必要的点, 从而进一步提高匹配速度。

2 多分辨逐步消除快速算法

2.1 多分辨逐步消除

对于简单的视频序列, 其时间和空间相关性很强。因此, 对某匹配点来说, 若在较低分辨率上式(11)不成立, 则有理由认为该匹配点在全分辨率上不等式(11)也不成立, 该匹配点即被消除, 否则, 若在较低分辨率上式(11)成立, 则再在较高分辨率上进一步判断。这就是多分辨逐步消除的基本思想。

对当前帧中的 $N \times N$ 大小的块 B , 要在参考帧中求其最优匹配块, 通过抽取降样, 分四种分辨率进行。为此, 记 $B_{o,o}$ 为将 B 按行和列均奇抽取的部分, $B_{o,e}$ 为将 B 按行奇抽取和列偶抽取的部分, $B_{e,o}$ 为将 B 按行偶抽取和列奇抽取的部分, $B_{e,e}$ 为将 B 按行和列均偶抽取的部分, 则 $B_{o,o}, B_{o,e}, B_{e,o}$ 和 $B_{e,e}$ 构成 B 的一个不重叠分割, 且:

$$B = B_{o,o} \cup B_{o,e} \cup B_{e,o} \cup B_{e,e}$$

消除过程中的四种分辨率分别定义为: $B_{o,o}, B_{o,o} \cup B_{o,e}, B_{o,o} \cup B_{o,e} \cup B_{e,o}$ 和 B 。在匹配过程中, 对 $R, Norm$ 和 SAD 分别用下标 1, 2, 3, 4 表示其对应于 $B_{o,o}, B_{o,e}, B_{e,o}, B_{e,e}$ 抽取下的计算量, 如 $R_1 = \sum_{(i,j) \in B_{o,o}} |I_t(i, j)|$, $Norm_3(x, y) = \sum_{(i,j) \in B_{e,o}} |I_{t-1}(i+x, j+y)|$, $SAD_4(x, y) = \sum_{(i,j) \in B_{e,e}} |I_{t-1}(i, j) - I_{t-1}(i+x, j+y)|$ 。设 S 为欲匹配求优的点集, (m, n) 为当前的最优点, 则多分辨逐步消除过程如下。

对任一 $(x, y) \in S$,

$$1) \text{ 计算 } Norm_1(x, y) = \sum_{(i,j) \in B_{o,o}} |I_{t-1}(i+x, j+y)|;$$

$$2) \text{ 若 } |Norm_1(x, y) - R_1| < SAD_1(m, n), \text{ 则:}$$

$$\text{计算 } Norm_2(x, y) = \sum_{(i,j) \in B_{o,e}} |I_{t-1}(i+x, j+y)|;$$

$$\text{若 } \sum_{i=1}^2 |Norm_i(x, y) - R_i| < \sum_{i=1}^2 SAD_i(m, n), \text{ 则:}$$

$$\text{计算 } Norm_3(x, y) = \sum_{(i,j) \in B_{e,o}} |I_{t-1}(i+x, j+y)|;$$

$$\text{若 } \sum_{i=1}^3 |Norm_i(x, y) - R_i| < \sum_{i=1}^3 SAD_i(m, n), \text{ 则:}$$

$$a) \text{ 计算 } Norm_4(x, y) = \sum_{(i,j) \in B_{e,e}} |I_{t-1}(i+x, j+y)|;$$

b) 若 $\sum_{i=1}^4 |Norm_i(x, y) - R_i| < \sum_{i=1}^4 SAD_i(m, n)$, 则:

计算 $SAD(x, y) = \sum_{(i,j) \in B} |I_{t-1}(i, j) - I_{t-1}(i+x, j+y)|$;

若 $SAD(x, y) < SAD(m, n)$, 则 $(m, n) = (x, y)$ 。

2.2 快速匹配运动估计

假设搜索范围是以原块的最初位置为中心, $w = \pm 7$ (即搜索半径) 的矩形窗, 若每个点逐一计算, 对较简单的视频序列, 显然是不必要的。NTSS^[3] 是一种简单有效的快速算法, 其估计的精度与 FS 十分接近, 但其速度提高了十几倍。基于 NTSS 思想, 本文的多分辨逐步消除快速匹配算法的基本步骤如下。其中, 对于两像素点: (x, y) 和 (x', y') , 定义它们之间的距离为:

$$D((x, y), (x', y')) = \max\{|x - x'|, |y - y'|\}$$

同时, 为了方便, 用 $S_k(x, y)$ 表示以 (x, y) 为中心, 和 (x, y) 距离为 k 的 8 各相邻像素坐标之集。

第一步: 由运动场的时间和空间相关性估计初始运动矢量, 其结果记为 (x_1, y_1) ;

第二步: 求 $(0, 0)$ 与 (x_1, y_1) 二者较优者, 记为 (x_2, y_2) ;

第三步: 多分辨逐步消除求优

1) 对 $S_1(x_2, y_2) \cup S_4(0, 0)$ 进行多分辨逐步消除求优, 其结果记为 (x_3, y_3) ;

2) 缩小搜索范围

a) 若 $d((x_2, y_2), (x_3, y_3)) = 0$, 则转 3);

b) 若 $d((x_2, y_2), (x_3, y_3)) = 1$, 则对 $S_1(x_3, y_3)$ 进行匹配(非消除法)求优, 转 3);

c) 若 $d((x_2, y_2), (x_3, y_3)) > 1$:

对 $S_2(x_3, y_3)$ 进行多分辨逐步消除求优, 其结果记为 (x_4, y_4) ;

对 $S_1(x_4, y_4)$ 进行匹配(非消除法)求优。

3) 算法结束。

算法的第一步是估计初始运动矢量, 这对逐步消除算法尤为重要, 较精确的初始运动矢量有利于消除更多的点, 从而提高运动估计的速度。对于较简单的视频序列, 由于运动场的时间和空间相关性较强, 本文通过对当前块的左邻块、上邻块和左上角邻块的已估计的运动矢量以及前一帧的当前位置块的运动矢量求均值来估计初始运动矢量。

3 实验结果

表 1 Akyio 序列的实验结果

	FS	NTSS	TSS	MSEA
PSNR	41.2482	41.2482	41.2482	41.2482
加法	115 200	8 046	12 800	3 846
绝对值	57 600	4 023	6 400	2 782

表 2 News 序列的实验结果

	FS	NTSS	TSS	MSEA
PSNR	36.2914	36.2786	36.2639	36.2786
加法	115 200	8 138	12 800	4 662
绝对值	57 600	4 069	6 400	3 300

在数值实验中, 对 QCIF 测试序列, 给出了不同运动估计算法, 其前 50 帧实验补偿后的平均峰值信噪比 (PSNR)、运动估计过程中平均每块的加法和绝对值运算次数。实验结果见

表 1 ~ 表 4。

表 3 Silence 序列的实验结果

	FS	NTSS	TSS	MSEA
PSNR	34.2826	34.1739	34.1945	34.1802
加法	115 200	8 421	12 800	6 291
绝对值	57 600	4 210	6 400	4 336

表 4 Foreman 序列的实验结果

	FS	NTSS	TSS	MSEA
PSNR	30.2974	30.1580	30.0847	30.1742
加法	115 200	10 373	12 800	10 223
绝对值	57 600	5 186	6 400	6 637

通过实验结果可以看出, 对于较简单的视频序列象 Akyio、News 和 Silence 等, 本文的 MSEA 方法, 其平均峰值信噪比和 NTSS 的实验结果不相上下, 但其运算量有了进一步的降低, 是 NTSS 运算量的 2/3 到 1/2 左右。对较复杂的视频序列象 Foreman 等, 本文的 MSEA 方法, 其平均峰值信噪比比 NTSS 的实验结果略有提高, 而运算量减少不多, 甚至有所增加。

基于 ITU-T LBC 专家组的 TMN5 测试模型, 本文将多分辨逐步消除算法嵌入在 TMN-codec 中, 得到基于快速运动估计的 H. 263 编码器, 并对 QCIF 格式的视频测试序列进行实验。在实验结果中 FS、MSEA 分别表示按全搜索算法和多分辨逐步消除算法进行编码。实验结果如图 1 和图 2 所示。

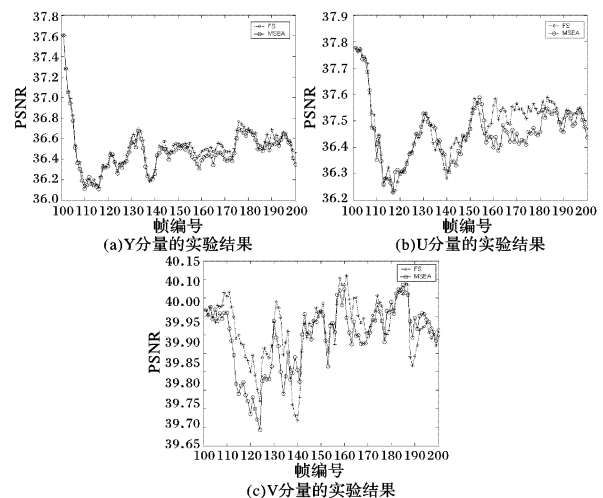


图 1 Akyio 序列实验的峰值信噪比 (帧频每秒 30 帧, 码率 184Kbps)

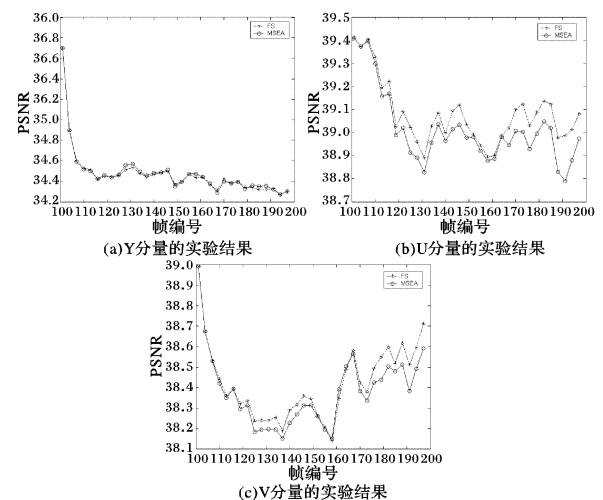


图 2 Container 序列的峰值信噪比 (帧频是每秒 10 帧, 码率 80Kbps)

(下转第 450 页)

从图 5 和图 6 可以看出,两种方法得到的结果完全相同,但经典 FFT 算法的计算过程花费时间 9.40 s,FS-FFT 只用了 8.72 s,效率提高约 7.23%。

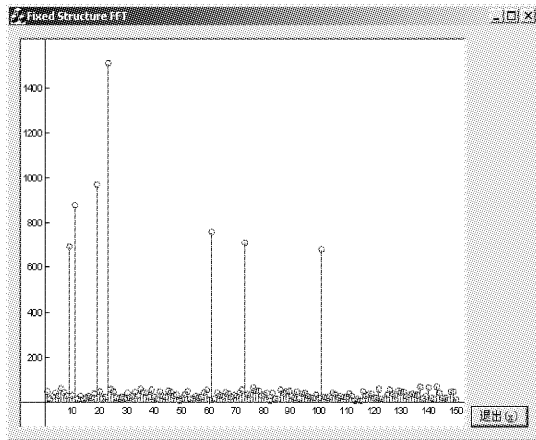


图 6 FS-FFT 运算结果(截取最重要的部分)

结合 NCG2050-A 的机械特性对图 5 和图 6 进行分析,我们发现频率 $f = 22\text{Hz}$ 是不正常的。通过查阅机械部分的设计图纸,发现该系统中只有一个齿轮会产生 22Hz 的机械振动频率,该齿轮位于磨削砂轮纵向伺服电机的变速箱中,因此推断是该齿轮出了问题。拆开该变速箱,果然发现该齿轮的一个轮齿意外断裂,换上新的齿轮后,系统开始正常运行。由于这种分析方法具有很强的针对性,避免了大规模拆卸系统和盲目查找故障带来的一系列不良后果。

4 结语

通过改变 FFT 算法中的蝶形结构,我们构建出一种固定结构的改进算法 FS-FFT。蝶形在运算过程中的每一级均保持不变,因此存取中间运算结果时不需要进行复杂的寻址操

作,从而简化运算步骤,提高处理速度,在实时性要求较高的场合,如雷达信号的快速处理等,具有一定的理论价值和实用价值。本文所提方法具有良好的扩展性,蝶形数目和循环次数由具体 n 值决定。进一步探讨蝶形的特性、构建性能更好的算法是下一步要做的工作。

参考文献:

- [1] 姜建国,曹建中,高玉明. 信号与系统分析基础 [M]. 北京: 清华大学出版社, 1997. 7
- [2] LIGUORI C, PAOLILLO A. Implementing Uncertainty auto-evaluation Capabilities on an Intelligent FFT Analyzer [J]. IEEE Trans. on Instrumentation and Measurement, 2004, 53(3): 700 - 708.
- [3] BOUGUEZEL S, AHMAD MO, SWAMY MNS. A Note on "Split Vector-Radix-2/8 2-D Fast Fourier Transform" [J]. IEEE Signal Processing Letters, 2005, 12(3): 185 - 188.
- [4] LIGUORI C, PAOLILLO A, PIGNOTTI A. An Intelligent FFT Analyzer with Harmonic Interference Effect Correction and Uncertainty Evaluation[J]. IEEE Trans. on Instrumentation and Measurement, 2004, 53(4): 1125 - 1131.
- [5] 吴镇扬. 数字信号处理的原理与实现 [M]. 南京: 东南大学出版社, 2002. 9
- [6] 余娟,王典洪,唐伯良. 一种结合 DFT 和 FFT 的实用傅里叶变换算法 [J]. 计算机应用与软件, 2005, 22(5): 38 - 39.
- [7] 尚明生,王庆先,孙世新. 集群系统中 BSP 模型上的并行 FFT 设计 [J]. 计算机应用, 2002, 22(7): 34 - 35, 38.
- [8] FAN G - X, LIU Q - H. Fast Fourier Transform for Discontinuous Functions [J]. IEEE Trans. on Antennas and Propagation, 2004, 52(2): 461 - 485.
- [9] 王立宁,乐光新,詹菲. MATLAB 与通信仿真 [M]. 北京: 人民邮电出版社, 2000.
- [10] CHAPMAN D. Sams Teach Yourself Visual C++ 6 in 21 Days [M]. Beijing: Tsinghua University Press, 2001. 7

(上接第 438 页)

4 结语

本文充分利用运动场在时间和空间上的相关性以及像素的空间相关性,给出了在不同匹配点上采用不同降样法的快速匹配算法,即多分辨逐步消除快速匹配算法,实验结果显示,对较简单的视频序列,这种方法取得了良好的效果,其平均峰值信噪比界于 TSS 和 NTSS 算法之间,但是,运算量有了大幅度的下降。

参考文献:

- [1] JAIN JR, JAIN AK. Displacement measurement and its application in interframe image coding [J]. IEEE Trans. Commun, Dec. 1981, COMM - 29: 1799 - 1908.
- [2] COGA KIT, HIRANO A, IJIMA Y, et al. Motion compensation interframe coding for video conferencing [J]. in Proc. Nat. Telecom. Conf81, New Orleans, Dec. 1992, 28: 239 - 251.
- [3] LI R, ZENG B, LIU ML. A New three-step search algorithm for block motion estimation [J]. IEEE Trans. Circuit Syst. Video Technol, 1994, 4: 438 - 442.
- [4] PURI A, HANG HM, SCHILLING DL. An efficient block-matching algorithm for motion compensated coding [A]. Proc ICASSP [C]. 1987. 1063 - 1066.
- [5] CHANBARI M. The cross - search algorithm for motion estimation

- [J]. IEEE Trans. Commun. July 1990, 1001: 950 - 953.
- [6] CHEN MJ, CHEN LG, CHIUH TG. One-dimension full search motion estimation algorithm for video coding [J]. IEEE Trans. Circuit Syst. Video Technol, 1994, 4: 504 - 509.
- [7] PO LM, MA WC. A novel four-step search algorithm for fast block motion estimation [J]. IEEE Trans. Circuit Syst. Video Technol, June. 1996, 6: 313 - 317.
- [8] CHENG F-H, SUN S-N. New fast and efficient tow-step search algorithm for block motion estimation [J]. IEEE Trans. Circuit Syst. Video Technol., Oct. 1999, 9: 977 - 983.
- [9] LI W, SALARI E. Successive elimination algorithm for motion estimation [J]. IEEE Trans Image Processing, Jan. 1995, 4(1): 105 - 107.
- [10] JUNG S-M, SHIN S-C, BAIK H, et al. Nobel successive elimination algorithm for estimation of motion vectors [A]. Proceedings. International Symposium on Multimedia Software Engineering [C]. Taipei, Taiwan, 2000. 332 - 335.
- [11] WANG H-S, MERSEREAU RM. Fast algorithms for the estimation of motion vectors [J]. IEEE Trans Image Processing, March 1999, 8(3): 435 - 438.
- [12] BRUNIG M, NIEHSEN W. Fast full-search block matching [J]. IEEE Trans. Circuit Syst. Video Technol., Feb. 2001, 11(2): 241 - 247.