

基于 Hadamard 变换和矢量分割的快速搜索算法

乔 阳 潘志斌 乔瑞萍 李东平 蔡 骋

(西安交通大学电子与信息工程学院, 西安 710049)

摘 要 为了对图像信号进行快速有效压缩,提出了一种改进的基于 Hadamard 变换和矢量分割的双测试算法。该算法首先在 Hadamard 域中对空域双测试算法的两个删除准则进行了等效变换;然后在实验的基础上,对变换域码字和输入矢量进行了恰当的矢量分割;最后在码字搜索过程中,利用其中一个最为有效的删除准则来排除大部分的不匹配码字。实验结果表明,改进算法能大大提高码字的搜索效率,搜索范围可减少到原始算法搜索范围的约 14% ~ 17%,总体编码时间也减少到原始算法编码时间的约 35% ~ 45%。

关键词 矢量量化 码字快速搜索 Hadamard 变换 单测试算法 双测试算法

中图法分类号: TN919. 8 文献标识码: A 文章编号: 1006-8961(2009)11-2269-07

An Improved Fast Search Algorithm Based on Hadamard Transform and Vector Partition

QIAO Yang, PAN Zhi-bin, QIAO Rui-ping, LI Dong-ping, CAI Cheng

(School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049)

Abstract An improved double test algorithm in Hadamard transform(HT) domain is proposed in this paper. First, two kick-out conditions in original DTA method are rewritten properly in HT domain. Then, codewords and input vectors in HT domain are properly partitioned based on the result of the preliminary experiment. Finally, by using the better kick-out condition many unmatched codewords can be eliminated. The simulation shows that the search space of the improved algorithm in this paper is reduced to about 14% ~ 17%, and the encoding time is reduced to about 35% ~ 45% compared to the original algorithm.

Keywords vector quantization, fast codeword search, Hadamard transform, single test algorithm, double test algorithm

1 引 言

矢量量化(vector quantization, VQ)^[1]作为一种高效的有损压缩技术,已经广泛应用于图像信号处理和语音信号处理等领域,但是矢量量化编码过程的计算量非常大,严重限制了矢量量化技术的应用和进一步发展,因此研究矢量量化中码字的快速搜索算法具有十分重要的意义。

矢量量化的码字搜索过程可描述为:在给定码

书 $C = \{y_0, y_1, \dots, y_{N-1} \mid y_i \in \mathbf{R}^k, i = 0, 1, \dots, w, \dots, N-1\}$ 中搜索与输入矢量 $x = (x_0, x_1, \dots, x_{k-1})^T$ 最匹配的码字(winner codeword) $y_w = (y_{w,0}, y_{w,1}, \dots, y_{w,k-1})^T$,以便使得 x 和 y_w 之间的失真最小。由于通常失真测度采用欧氏距离衡量,即

$$d(x, y_i) = \sum_{l=0}^{k-1} (x_l - y_{i,l})^2 \quad (1)$$

因此可通过下式来计算最佳匹配码字

$$d(x, y_w) = \min_{y_i \in C} d(x, y_i) \quad (2)$$

如果采用穷尽搜索(full search, FS)算法,那么

对每一个输入矢量要获得最佳匹配码字 y_w 就需要计算 Nk 次乘法、 $N(2k-1)$ 次加法和 $N-1$ 次比较运算,其计算量非常大。为了减小 FS 搜索算法的计算量,许多文献提出了各种快速算法,如部分失真搜索 (partial distortion search, PDS) 算法^[2]、双测试算法 (double test algorithm, DTA)^[3]、等均值最近邻搜索 (equal-average nearest neighbor search, ENNS) 算法^[4-5] 及其改进算法、基于小波变换的搜索 (wavelet transform search, WTS) 算法^[6] 和基于 Hadamard 变换的快速搜索 (Hadamard transform search, HTS) 算法^[7-8] 等等,这些算法可不同程度地提高码字的搜索效率。本文在 Hadamard 域中对 DTA 算法进行了进一步的改进。

2 单测试算法和双测试算法

在空域 (spatial domain) 中,文献 [3] 通过对平方误差的展开式进行分析,提出了双测试算法。因为式 (1) 定义的欧氏距离测度也可以写成下面的形式

$$\begin{aligned} d(\mathbf{x}, \mathbf{y}_i) &= \sum_{l=0}^{k-1} x_l^2 + \sum_{l=0}^{k-1} y_{i,l}^2 - 2 \sum_{l=0}^{k-1} x_l y_{i,l} \\ &= \|\mathbf{x}\|^2 + \|\mathbf{y}_i\|^2 - 2 \sum_{l=0}^{k-1} x_l y_{i,l} \end{aligned} \quad (3)$$

当 $x_l \geq 0$, 且 $y_{i,l} \geq 0$ 时,显然有

$$\sum_{l=0}^{k-1} x_l y_{i,l} \leq x_{\max} \sum_{l=0}^{k-1} y_{i,l} \quad (4)$$

其中, x_{\max} 是矢量 \mathbf{x} 的所有分量中的最大值。

根据式 (3) 和式 (4), 文献 [3] 定义了一个码字排除的测试准则, 如果

$$d_1(\mathbf{x}, \mathbf{y}_i) = \|\mathbf{x}\|^2 + \|\mathbf{y}_i\|^2 - 2x_{\max} \sum_{l=0}^{k-1} y_{i,l} \quad (5)$$

那么 $d_1(\mathbf{x}, \mathbf{y}_i)$ 一定满足

$$d_1(\mathbf{x}, \mathbf{y}_i) \leq d(\mathbf{x}, \mathbf{y}_i) \quad (6)$$

设 d_{\min} 为码字搜索过程中已经找到的当前最小失真, 在码字测试过程中, 若 $d_1(\mathbf{x}, \mathbf{y}_i) \geq d_{\min}$ 成立, 则必然有 $d(\mathbf{x}, \mathbf{y}_i) \geq d_{\min}$ 成立, 这样不必计算欧氏距离 $d(\mathbf{x}, \mathbf{y}_i)$ 就可排除当前码字, 转而搜索下一个码字; 否则使用 PDS 算法进行当前码字能否被排除的测试, 并当 $d(\mathbf{x}, \mathbf{y}_i) < d_{\min}$ 时, 更新 d_{\min} 。因为它只利用式 (6) 进行一次测试, 故这一算法被称为单测试算法 (single test algorithm, STA)。

另外, 根据对称性原则, 文献 [3] 中还定义了类

似的另一个码字排除的测试准则, 即

$$d_2(\mathbf{x}, \mathbf{y}_i) = \|\mathbf{x}\|^2 + \|\mathbf{y}_i\|^2 - 2(y_i)_{\max} \sum_{l=0}^{k-1} x_l \quad (7)$$

其中, $(y_i)_{\max}$ 为码字 \mathbf{y}_i 的所有分量中的最大值。实际应用时, 可以结合以上两个失真测度来进行码字删除判断。通常, 先使用测试准则 $d_1(\mathbf{x}, \mathbf{y}_i)$ 来判断, 如果 $d_1(\mathbf{x}, \mathbf{y}_i)$ 不能排除当前码字时, 再使用 $d_2(\mathbf{x}, \mathbf{y}_i)$ 进行判断, 因为利用式 (5) 和式 (7) 进行了两次测试, 所以这一算法称为双测试算法。

STA 算法的效率取决于判断条件 $d_1(\mathbf{x}, \mathbf{y}_i) \geq d_{\min}$ 的成立概率, 而 DTA 算法的效率则取决于判断条件 $d_1(\mathbf{x}, \mathbf{y}_i) \geq d_{\min}$ 或者 $d_2(\mathbf{x}, \mathbf{y}_i) \geq d_{\min}$ 的成立概率, 若这一概率越高, 则算法的效率越高。根据实验结果可知, 若使用大小为 512, 维数为 4×4 的码书, 测试图像为 512×512 大小的 Lena 图像, 则使用 STA 和 DTA 算法时, 这一概率分别为 55% 和 66%, 即这两个算法的搜索范围只能分别减小到 FS 算法的 45% 和 34%, 考虑到 $d_1(\mathbf{x}, \mathbf{y}_i)$ 和 $d_2(\mathbf{x}, \mathbf{y}_i)$ 的计算负荷, 其搜索效率仍然不高, 有必要对其做进一步的改进。

3 本文的算法

3.1 Hadamard 变换的基本定义及定理^[7]

定义 1

假定 \mathbf{H}_n 为 $2^n \times 2^n$ 大小的 Hadamard 矩阵, 其元素属于集合 $\{1, -1\}$, 那么 Hadamard 矩阵可按下面的方式递归产生:

$$\mathbf{H}_1 = \begin{Bmatrix} 1 & 1 \\ 1 & -1 \end{Bmatrix}, \quad \mathbf{H}_{n+1} = \begin{Bmatrix} \mathbf{H}_n & \mathbf{H}_n \\ \mathbf{H}_n & -\mathbf{H}_n \end{Bmatrix}$$

则 k 维 ($k=2^n$) 矢量 \mathbf{x} 经过 Hadamard 变换后得到的矢量 \mathbf{X} 为

$$\mathbf{X} = \mathbf{H}_n \mathbf{x}$$

并且 \mathbf{x} 和 \mathbf{X} 之间服从如下定理给出的关系。

定理 1

k 维矢量 \mathbf{x} 的 Hadamard 变换后的矢量 \mathbf{X} 的能量是其空域内能量的 k 倍, 即

$$(\|\mathbf{X}\|_2)^2 = k(\|\mathbf{x}\|_2)^2$$

证明:

$$\begin{aligned} (\|\mathbf{X}\|_2)^2 &= \mathbf{X}^T \mathbf{X} = (\mathbf{H}_n \mathbf{x})^T (\mathbf{H}_n \mathbf{x}) = \mathbf{x}^T \mathbf{H}_n^T \mathbf{H}_n \mathbf{x} \\ &= \mathbf{x}^T k \mathbf{I}_k \mathbf{x} = k \mathbf{x}^T \mathbf{x} = k(\|\mathbf{x}\|_2)^2 \end{aligned}$$

这里 \mathbf{I}_k 为 $k \times k$ 维的单位矩阵。

3.2 本文算法的原理

根据前面的实验结果,空域中直接使用双测试算法的搜索效率并不高,为了提高该算法的搜索效率,考虑对测试准则进行适当的变形,以使得失真 $d_1(\mathbf{x}, \mathbf{y}_i)$ 的值变得更大,使其更接近欧氏距离失真 $d(\mathbf{x}, \mathbf{y}_i)$,以便排除更多的不匹配码字,即要设法尽量减小式(5)中 $x_{\max} \sum_{l=0}^{k-1} y_{i,l}$ 这一项的值,以有利于码字的排除。其中一种可行的方法是对原始的 k 维矢量进行子矢量分割,通常可以用输入矢量 \mathbf{x} 的前 h 维 ($1 \leq h \leq k$) 分量构成第 1 个子矢量 \mathbf{x}^f (上角 f 代表 first), 剩余分量构成第 2 个子矢量 \mathbf{x}^s (上角 s 代表 second), 那么一定有

$$\sum_{l=0}^{k-1} x_l y_{i,l} \leq \left(x_{\max}^f \sum_{l=0}^{h-1} y_{i,l} + x_{\max}^s \sum_{l=h}^{k-1} y_{i,l} \right) \leq x_{\max} \sum_{l=0}^{k-1} y_{i,l} \quad (8)$$

这里 x_{\max}^f 为第 1 个子矢量 \mathbf{x}^f 所有分量中的最大值, x_{\max}^s 为第 2 个子矢量 \mathbf{x}^s 所有分量中的最大值。

这样,就可以将式(5)中的 $x_{\max} \sum_{l=0}^{k-1} y_{i,l}$ 项用 $\left(x_{\max}^f \sum_{l=0}^{h-1} y_{i,l} + x_{\max}^s \sum_{l=h}^{k-1} y_{i,l} \right)$ 代替来得到更大的 $d_1(\mathbf{x}, \mathbf{y}_i)$ 的值,以增强式(5)对非匹配码字的排除能力。显然,如何通过选取 h 的值来进行子矢量分割是一个关键问题。

若使用大小为 512, 维数为 4×4 的码书, 测试图像为 512×512 大小的 Lena 图像, 则实验证明(图 1 所示), 无论子矢量分割的维数 h 如何选取, 在

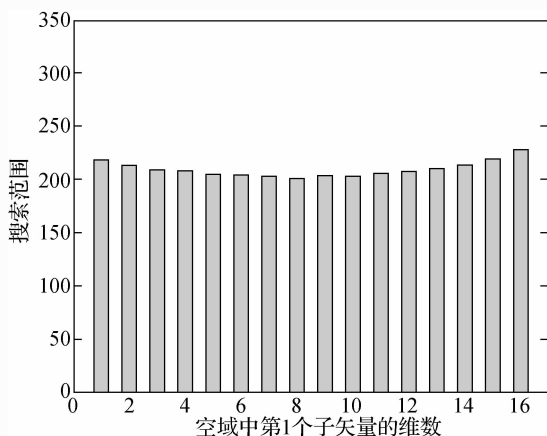


图 1 空域中第 1 个子矢量的维数与搜索范围的关系

Fig. 1 Relationship between the dimension of first subvector and search space in spatial domain

空域内进行子矢量分割仅会使搜索范围得到很小的下降。当 $h = k$ 时, $\mathbf{x}^f = \mathbf{x}, \mathbf{x}^s = 0$ 相当于没有进行子矢量分割。但根据式(8), 对矢量进行一次分割就会增加 1 次乘法和 1 次加法运算, 这样总的计算量并没有明显减少。

因为空域中的矢量在各维上的能量分布均匀, 且矢量分割对于搜索范围减小的效果不明显, 所以可以考虑在变换域中对矢量进行分割。由于 Hadamard 变换可使变换域矢量的能量集中在前面的少数几维分量上, 且矢量在 Hadamard 变换前后的能量呈固定的 k 倍关系, 因此在变换域内搜索最匹配码字和在空域内搜索最匹配码字是等价的, 这样就有可能在 Hadamard 变换域内进行更加有效的矢量分割。

在 Hadamard 变换域中, 欧氏距离测度可展开成下面的形式:

$$D(\mathbf{X}, \mathbf{Y}_i) = \sum_{l=0}^{k-1} X_l^2 + \sum_{l=0}^{k-1} Y_{i,l}^2 - 2 \sum_{l=0}^{k-1} X_l Y_{i,l} = \|\mathbf{X}\|^2 + \|\mathbf{Y}_i\|^2 - 2 \sum_{l=0}^{k-1} X_l Y_{i,l} \quad (9)$$

其中 \mathbf{X} 和 \mathbf{Y}_i 分别是输入矢量 \mathbf{x} 和码书中码字 \mathbf{y}_i 经过 Hadamard 变换后的矢量, 那么 $D(\mathbf{X}, \mathbf{Y}_i) = kd(\mathbf{x}, \mathbf{y}_i)$ 成立。

如果矢量 \mathbf{x}, \mathbf{y}_i 经 Hadamard 变换后的矢量 \mathbf{X}, \mathbf{Y}_i 的分量出现负值, 则不等式 $\sum_{l=0}^{k-1} X_l Y_{i,l} \leq X_{\max} \sum_{l=0}^{k-1} Y_{i,l}$ 在 Hadamard 域中就不成立了, 故空域中的 STA 算法不能直接用到 Hadamard 域中。为了解决变换后矢量 \mathbf{X}, \mathbf{Y}_i 的分量为负值的问题, 文献[3]中提出的一种解决方案是对变换后的矢量加一个正的偏移量, 以便使变换后矢量的全部分量都变为正值, 这个正的偏移量的最小值可以取变换后矢量 \mathbf{X} 和码字 \mathbf{Y}_i 的负的最小分量的绝对值, 但是这种方法的计算量很大, 因为对每个输入的变换后的矢量 \mathbf{X} 不仅需要在线计算 N 个偏移量, 还要在线对所有码字增加偏移量; 另外, 也可以取最大可能的偏移量 ($255 \times k/2$), 但实验证明采用这种方法搜索范围几乎没有下降。与以上两种方法不同, 本文对 Hadamard 变换后的矢量 \mathbf{X}, \mathbf{Y}_i 取绝对值, 即相当于对正的分量没有偏移, 对负的分量偏移了最小偏移量的两倍, 这不仅可减少计算量, 也保证了搜索效率。这样式(4)在 Hadamard 域中可写成下面的形式

$$\sum_{l=0}^{k-1} X_l Y_{i,l} \leq |\mathbf{X}|_{\max} \sum_{l=0}^{k-1} |Y_{i,l}| \quad (10)$$

其中, $|\mathbf{X}|_{\max}$ 是矢量 \mathbf{X} 中所有分量的绝对值中的最

大值。

那么,结合式(9)和式(10),可以得出

$$D_1(\mathbf{X}, \mathbf{Y}_i) = \|\mathbf{X}\|^2 + \|\mathbf{Y}_i\|^2 - 2 \|\mathbf{X}\|_{\max} \sum_{l=0}^{k-1} |Y_{i,l}| \quad (11)$$

式(11)为本文提出的 Hadamard 域中的单测试算法(HSTA)。

那么,接下来的问题是如何在 Hadamard 域内进行有效的矢量分割。为了提高码字删除效率,就必须增大式(11)的值,即减小 $\|\mathbf{X}\|_{\max} \sum_{l=0}^{k-1} |Y_{i,l}|$ 的值,通常可利用矢量分割来减小其值。另外,如何选择矢量分割的维数也是一个关键问题。图 2 显示了 Hadamard 域中第 1 个子矢量的维数与搜索范围的关系,本文算法采用了 Hadamard 域的单测试算法(HSTA),测试条件如前。由图 2 可以看出,不同的子矢量分割方法对搜索空间的影响非常明显,其中第 1 维分量单独组成一个子矢量,剩下的 $k-1$ 维组成另一个子矢量的矢量划分是最优的,此时的搜索范围最小,仅为 35 个码字,即搜索范围可以减小到 FS 算法的 6.8%。这和 Hadamard 变换能够使能量主要集中在矢量的第 1 维分量上相一致。

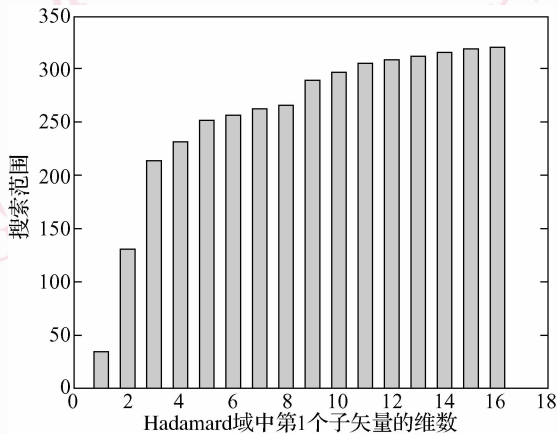


图 2 Hadamard 域中第 1 个子矢量的维数与搜索范围的关系

Fig. 2 Relationship between the dimension of first subvector and search space in Hadamard domain

至此,引入子矢量分割后的测试准则式(11)在 Hadamard 域里最终可以写成下面的形式

$$D_1(\mathbf{X}, \mathbf{Y}_i) = \|\mathbf{X}\|^2 + \|\mathbf{Y}_i\|^2 - 2 \left(X_0 Y_{i,0} + \|\mathbf{X}\|_{\max} \sum_{l=1}^{k-1} |Y_{i,l}| \right) \quad (12)$$

其中 $\|\mathbf{X}\|_{\max}$ 为输入变换后矢量的第 2 个子矢量中所有分量的绝对值中的最大值。

同样,另一个测试准则(式(7))在 Hadamard 域里最终可以写成下面的形式:

$$D_2(\mathbf{X}, \mathbf{Y}_i) = \|\mathbf{X}\|^2 + \|\mathbf{Y}_i\|^2 - 2 \left(X_0 Y_{i,0} + \|\mathbf{Y}_i\|_{\max} \sum_{l=1}^{k-1} |X_l| \right) \quad (13)$$

其中 $\|\mathbf{Y}_i\|_{\max}$ 为码字的第 2 个子矢量中所有分量的绝对值中的最大值。

另外,由于在计算输入矢量与码字失真的过程中, $\|\mathbf{X}\|^2$ 是一个常量,所以可以考虑采用等价的误差测度 d^H (上角 H 代表 Hadamard 域),即

$$d^H = D(\mathbf{X}, \mathbf{Y}) - \|\mathbf{X}\|^2 \quad (14)$$

这样式(12)、式(13)可进一步写成如下的等价形式

$$d_1^H = D_1(\mathbf{X}, \mathbf{Y}_i) - \|\mathbf{X}\|^2 = \|\mathbf{Y}_i\|^2 - 2 \left(X_0 Y_{i,0} + \|\mathbf{X}\|_{\max} \sum_{l=1}^{k-1} |Y_{i,l}| \right) \quad (15)$$

$$d_2^H = D_2(\mathbf{X}, \mathbf{Y}_i) - \|\mathbf{X}\|^2 = \|\mathbf{Y}_i\|^2 - 2 \left(X_0 Y_{i,0} + \|\mathbf{Y}_i\|_{\max} \sum_{l=1}^{k-1} |X_l| \right) \quad (16)$$

根据上面的叙述,当规定 L 为输入序列的长度时, d_{\min} 为当前已经找到的最小的欧氏距离,那么 Hadamard 变换域中单测试算法的流程如下:

离线计算并存储所有码字的 Hadamard 变换后的码字 \mathbf{Y}_i
 离线计算并存储所有码字的范数 $\|\mathbf{Y}_i\|^2$ 以及 $2Y_{i,0}$ 和

$$2 \sum_{l=1}^{k-1} |Y_{i,l}|$$

for $i = 0$ to $L - 1$

在线计算输入矢量的 Hadamard 变换后的矢量 \mathbf{X} , $\|\mathbf{X}\|^2$ 和 $\|\mathbf{X}\|_{\max}$

$$d_{\min} = \infty$$

for $j = 0$ to $N - 1$

$$d = d_1^H;$$

if $(d > d_{\min})$ continue;

$$d = d_H;$$

if $(d > d_{\min})$ continue;

index = j ;

$$d_{\min} = d;$$

next j

next i

同样, Hadamard 变换域中的双测试算法(HDTA)的流程可以描述如下:

离线计算并存储所有码字的 Hadamard 变换后的码字 \mathbf{Y}_i

离线计算并存储所有码字的范数 $\|Y_i\|^2, 2Y_{i,0}, 2\sum_{l=1}^{k-1} |Y_{i,l}|$ 和

$2|Y_i^s|_{\max}$
for $i=0$ to $L-1$

在线计算输入矢量的 Hadamard 变换后的矢量 $X, \|X\|^2,$

$|X^s|_{\max}$ 和 $\sum_{l=1}^{k-1} |X_l|$

$d_{\min} = \infty$

for $j=0$ to $N-1$

$d = d_j^H;$

if ($d > d_{\min}$) continue;

$d = d_j^H;$

if ($d > d_{\min}$) continue;

$d = d_H;$

if ($d > d_{\min}$) continue;

index = $j;$

$d_{\min} = d;$

next j

next i

在算法复杂度方面,本文算法和原始算法具有相同的空间复杂度,原始算法需要离线存储空域码书、码字的范数、和值和最大值,而本文算法则需要离线存储变换域的码书、码字的范数、第 2 个子矢量的和值以及最大值。在时间复杂度方面,在线计算时,本文算法只比原始算法多计算 1 次输入矢量的 Hadamard 变换,而且快速 Hadamard 变换的计算复杂度很小,只需要 $k \times \lg k$ 次加/减法,但由于本文算法相对原始算法可以明显地减小搜索范围,故总体上可极大地提高搜索效率。

4 模拟实验

为了验证本文算法的效果,采用了 512×512 大

表 1 原始算法和改进算法的搜索范围的比较

Tab. 1 Comparison of search space among the conventional algorithms and proposed algorithm

算法	Lena			Peppers			Airplane		
	256	512	1024	256	512	1024	256	512	1024
STA1 ^[3]	115	228	469	104	208	428	87	171	360
HSTA1	19	35	66	16	31	57	15	31	55
STA2 ^[3]	141	286	549	132	269	512	136	270	507
HSTA2	23	42	75	21	39	69	19	38	66
DTA ^[3]	86	176	357	79	162	328	66	135	279
HDTA	16	29	53	14	26	47	13	26	46

小的 Lena 图像作为训练集,先利用 LBG 算法生成码书,码书大小分别为 256, 512 和 1024, 矢量维数为 4×4 。测试图像为 512×512 的 Lena, Peppers 和 Airplane 图像。仿真实验在 Intel P4 主频 2.8 GHz、内存 512 M 的个人计算机上进行。

本文分别从搜索范围、运算量和编码时间 3 个方面与文献[3]中的算法进行了比较。表 1 给出了对每个输入矢量本文的改进算法和原始算法的平均搜索范围的比较,空域原始算法(STA1 和 STA2)分别使用式(5)和式(7)测试准则,而对应的 Hadamard 域改进算法(HSTA1 和 HSTA2)则分别使用式(15)和式(16)测试准则。从表 1 可以看出,本文提出的改进算法的搜索范围均可下降到空域中原始算法搜索范围的 14% ~ 17% 左右。另外,在空域和 Hadamard 变换域中,第 1 个测试准则的算法——STA1, HSTA1 均比第 2 个测试准则的算法——STA2, HSTA2 更加有效,这是因为对于每个输入矢

量,测试准则中,与输入矢量有关的部分 $(x_{\max}, \sum_{l=0}^{k-1} x_l, |X^s|_{\max}$ 和 $\sum_{l=1}^{k-1} |X_l|)$ 为常量,所以测试准则的效率主要取决于与码字有关的那一部分 $((y_i)_{\max}, \sum_{l=0}^{k-1} y_{i,l}, |Y_i^s|_{\max}$ 和 $\sum_{l=1}^{k-1} |Y_{i,l}|)$, 而其和值的分布则要比最大值的分布平稳,所以第 1 个测试准则要更加有效一些。因此在实际使用单测试算法时,可以使用第 1 个测试准则的算法——STA1, HSTA1, 而不考虑使用第 2 个准则的算法——STA2, HSTA2。

表 2 给出了对每个输入矢量,空域和 Hadamard 域中单测试算法 (STA1 和 HSTA1) 与双测试算法 (DTA 和 HDTA) 的平均计算量的比较,测试图像为 512×512 大小的 Lena 图像。从表 2 可以看出,本文提出的改进算法大大降低了原始算法加法和比较的运算量,但乘法运算量并没有明显减小。这一方面是由于搜索范围的下降,改进的算法大大减小了使

用 PDS 算法计算欧氏距离时的乘法、加法和比较计算量,并且使用等价误差测度可减少大量的加法运算;另一方面,由于使用了子矢量分割的方法,对每个输入矢量,在使用测试条件进行判决的过程中,改进的算法要比原始算法多 N 次乘法和 N 次加法运算,所以乘法计算量几乎没有变化,而加法和比较运算量则得到很大下降。

表 2 原始算法和改进算法的计算量的比较

Tab. 2 Comparison of computational cost among the conventional algorithms and proposed algorithm

算法	256			512			1024		
	乘法	加法	比较	乘法	加法	比较	乘法	加法	比较
STA1 ^[3]	699	1 383	799	1 224	2 433	1 437	2 283	4 550	2 736
HSTA1	678	831	441	1 243	1 448	766	2 373	2 685	1 415
DTA ^[3]	759	1 519	830	1 360	2 720	1 520	2 565	5 131	2 906
HDTA	694	877	437	1 269	1 516	756	2 419	2 792	1 394

表 3 给出了 FS 算法、PDS 算法、空域中的原始算法 (STA1 和 DTA) 和 Hadamard 域中的改进算法 (HSTA1 和 HDTA) 的编码时间 (单位为 ms)。从表 3 可见,无论空域,还是 Hadamard 域,从编码时间看,单测试算法都比双测试算法有效,这是因为实际编码时,单测试算法不能排除的码字,双测试算法大多数也不能排除,故双测试算法相对于单测试算法

的搜索范围减小得很少,但由于双测试算法比单测试算法要多计算一次测试准则,故总体计算量无明显改善。本文的改进算法—HSTA1 的编码时间可减少到原始算法—STA1 编码时间的约 35% ~ 45%,可见,编码速度有了明显提高。综上所述,在实际使用中,建议使用以式 (15) 作为测试准则的 HSTA1 算法。

表 3 各算法编码时间的比较

Tab. 3 Comparison of coding time among the conventional algorithms and proposed algorithm

单位:ms

算法	Lena			Peppers			Airplane		
	256	512	1 024	256	512	1 024	256	512	1 024
FS	171	341	683	171	341	683	171	341	683
PDS	92	163	300	94	170	316	83	156	294
STA1 ^[3]	115	219	424	110	217	415	96	185	370
HSTA1	48	81	150	45	78	143	43	78	142
DTA ^[3]	117	227	448	110	218	431	95	191	387
HDTA	52	89	163	50	85	157	48	85	153

5 结 论

本文提出的改进算法是在 Hadamard 变换域内进行的,并结合子矢量分割的方法对 DTA 算法的两

个测试准则进行了等效变换。根据仿真实验的结果可知,使用 Hadamard 域内的单测试算法 (HSTA1) 进行码字搜索比较有效。本文的改进算法较原始算法可明显提高搜索效率,其搜索范围可减小到原始算法的 14% ~ 17% 左右,编码时间也减少到原始算

法的约 35% ~ 50%。

参考文献 (References)

- 1 Linde Y, Buzo A, Gray R M. An algorithm for vector quantizer design [J]. IEEE Transactions on Communications, 1980, **28**(1): 84-95.
- 2 Bei C D, Gray R M. An improvement of the minimum distortion encoding algorithm for vector quantization[J]. IEEE Transactions on Communications, 1985, **33**(10): 1132-1133.
- 3 Torres T, Huguet J. An improvement on codebook search for vector quantization [J]. IEEE Transactions on Communications, 1994, **42**(2): 208-210.
- 4 Guan L, Kamel M. Equal-average hyperplane partitioning method for vector quantization of image data [J]. Pattern Recognition Letters, 1992, **13**(10): 693-699.
- 5 Pan Z, Kotani K, Ohmi T. A unified projection method for fast search of vector quantization [J]. IEEE Signal Processing Letters, 2004, **11**(7): 637-640.
- 6 Hwang W J, Jeng S S, Leou M R. Fast codeword search algorithm using wavelet transform and partial distance search techniques [J]. Electronics Letters, 1997, **33**(5): 365-366.
- 7 Lu Z M, Pan J S, Sun S H. Efficient codeword search algorithm based on Hadamard transform [J]. Electronics Letters, 2000, **36**(16): 1364-1365.
- 8 Pan Z, Kotani K, Ohmi T. Enhanced fast encoding method for vector quantization by finding an optimally-ordered Walsh transform kernel [A]. In: Proceedings of IEEE International Conference on Image Processing [C], Genoa, Italy, 2005, **1**: 573-576.