第 3 卷 第 4 期
1998 年 4 月

中国图象图形学报
Journal of Image and Graphics

Vol. 3, No. 4
Apr. 1998

基于模糊矢量量化图象编码的研究*

张基宏　王　晖
（深圳大学信息工程学院，深圳　518060）

Yoshito Ueno
（日本创价大学工学部情报学科，东京 192）

摘　要　分析了模糊矢量量化(FVQ)图象编码的原理，给出了 FVQ 设计三要素。提出了用于图象编码的指数型模糊矢量量化算法(FVQE)。实验结果表明，FVQE 的图象编码性能与 FVQ 相当，但收敛速度要略快于 FVQ 算法。

关键词　图象编码，模糊矢量量化，模糊 K-均值算法

1　引　言

经典的 LBG 码书设计算法是基于训练矢量和码书矢量的最小平均失真度[1]。LBG 算法简单、易于实现，但它强烈依赖于初始码书的选取，并且容易陷入局部最小。选取一个好的初始码书可以用随机码、分裂、最近邻域等方法来实现，但局部最小问题似乎没有简单的解决方法。近年来，一些学者采用随机松弛技术使平均失真度最小以搜索全局最佳码书[2,3]。这些技术的共同点是：在每次搜索最小平均失真度的迭代过程中，对某种失真度或一定失真度下所包括的独立变量以随机形式进行干扰。随机松弛技术能产生全局最小码书，但计算量很大。

上述码书设计算法均是基于硬决定的，即每个训练矢量是根据一些准则分配给单个聚类，而忽略了该训练矢量属于其它聚类的可能性。模糊 K-均值(FKM)聚类算法为训练集的每个元素分配一个介于 0 和 1 之间的隶属值，表示训练矢量隶属于一个确定聚类的程度[4]。FKM 算法性能比 LBG 好，但很少用来设计码书，因为计算量很大。为此 Nicolas. B. K 等人提出了模糊矢量量化(FVQ)算法[5]。FVQ 设计的码书性能与 FCM 相当；运算量也小于 FCM，但仍然较大。

2　FVQ 码书设计算法原理分析

FVQ 将每一聚类作为一个模糊集，用隶属函数来表示训练矢量隶属于一个确定聚类的程度。这样每个训练矢量依隶属函数的测度分配给多个聚类。若被考虑的训练矢量是一超球体的中心，对存在重叠的超球体的中心保证了所有训练矢量参与其中。有效的降低了设计结果对初始码书的依赖性。FVQ 通过迭代，即逐步收敛重叠的超球体来实现训练矢量逐步从软决定向硬决定的转变。在 FVQ 算法的最后阶段，根据保证不会局部最小的限制条件，生成全局最佳的最终码书矢量。

为了使训练所得的码书对初始码书的依赖性小，并且不会陷入局部最小和运算量较低；FVQ 的设计必需遵循下面三要素。设训练矢量集 χ 是 n 维，大小为 M，$\chi=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_M\}$，$\boldsymbol{x}_i\in R^n\ \forall\ i=1,2,\cdots,M$；码书矢量集为 y，大小为 k，$y=\{\boldsymbol{y}_1,\boldsymbol{y}_2,\cdots,\boldsymbol{y}_k\}$，$\boldsymbol{y}_j\in R^n\ \forall\ j=1,2,\cdots,k$。

第一个要素是训练矢量从软决定向硬决定的转变原则。转变速度可利用聚类过程中超球体的收缩方案来调整，为了折衷转变速度和聚类质量，采用以

---

* 本文得到了广东省重点学科重点科研项目的资助(№ GDUSD 9703)
收稿日期：1997-06-23；收到修改稿日期：1997-09-15

下策略：令 $I_i^{(v)}$ 是在 v 次迭代中，属于中心位于训练矢量 $\boldsymbol{x}_i\in\chi$ 超球体的码书矢量集合。在 v 次迭代后，中心位于训练矢量 $\boldsymbol{x}_i$ 超球体所包含的码书矢量 $\boldsymbol{y}_i\in I_i^{(v)}$ 须满足对 $\boldsymbol{x}_i$ 的距离小于或等于 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j\in I_i^{(v)}$ 的平均距离

$$d_{ave}^{(v)}(\boldsymbol{x}_i)=\frac{1}{N(I_i^{(v)})}\sum_{\boldsymbol{y}_j\in I_i^{(v)}}d(\boldsymbol{x}_i,\boldsymbol{y}_j)\tag{1}$$

式中 $N(I_i^{(v)})$ 是集 $I_i^{(v)}$ 中的元素数目。这样就得到集，$I_i^{(v+1)}$ 的公式：

$$I_i^{(v+1)}=\{\boldsymbol{y}_i\in I_i^{(v)}:d(\boldsymbol{x}_i,\boldsymbol{y}_j)\leqslant d_{ave}^{(v)}(\boldsymbol{x}_i)\}\tag{2}$$

如果 $I_i^{(v)}$ 只包含一个码书矢量，即 $N(I_i^{(v)})<2$，则训练矢量 $\boldsymbol{x}_i$ 就从模糊状态转变为硬状态。

第二个要素是的隶属函数的选取。隶属函数 $\mu_j(\boldsymbol{x}_i)$ 表示训练矢量分配给第 j 个聚类的不确定度。$\mu_j(\boldsymbol{x}_i)j=1,2,\cdots,k$ 的计算依赖于在 v 次迭代后集 $I_i^{(v)}$ 所包含的聚类数目。若中心位于训练矢量 $\boldsymbol{x}_i$ 的超球体只包含一个码书矢量 $\boldsymbol{y}_{j^*}$，由第一个要素可知，$\boldsymbol{x}_i$ 已经被分配 j^* 个聚类中，对应的隶属函数值为1；即如果 $d(\boldsymbol{x}_i,\boldsymbol{y}_j)=d(\boldsymbol{x}_i,\boldsymbol{y}_{j^*})$，则 $\mu_j(\boldsymbol{x}_i)=1$；否则 $\mu_j(\boldsymbol{x}_i)=0$。若集 $I_i^{(v)}$ 包含超过一个的码书矢量，则值介于0和1之间。如果 $\boldsymbol{y}_j\notin I_i^{(v)}$，则 $\mu_j(\boldsymbol{x}_i)=0$；否则 $\boldsymbol{y}_j\in I_i^{(v)}$，隶属函数依赖于 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j\in I_i^{(v)}$ 之间的距离，即 $\mu_j(\boldsymbol{x}_i)=f(d(\boldsymbol{x}_i,\boldsymbol{y}_j),\boldsymbol{y}_j\in I_i^{(v)})$。这时必须满足以下3个条件：$\mu_j(\boldsymbol{x}_i)$ 是距离 $d(\boldsymbol{x}_i,\boldsymbol{y}_j)$ 的递减函数；当 $d(\boldsymbol{x}_i,\boldsymbol{y}_j)$ 趋于0时，$\mu_j(\boldsymbol{x}_i)$ 逼近1；当 $d(\boldsymbol{x}_i,\boldsymbol{y}_j)$ 趋于 $d_{\max}(\boldsymbol{x}_i)$ 时，$\mu_j(\boldsymbol{x}_i)$ 逼近0。其中 $d_{\max}(\boldsymbol{x}_i)$ 是训练矢量 $\boldsymbol{x}_i$ 与码书矢量 $\boldsymbol{y}_j\in I_i^{(v)}$ 之间的最大距离，定义为 $d_{\max}(\boldsymbol{x}_i)=\max\limits_{\boldsymbol{y}_j\in I_i^{(v)}}d(\boldsymbol{x}_i,\boldsymbol{y}_j)$。

第三个要素是判别门限值 ε 和 ε 的选择。令 $C^{(v)}$ 为在 v 次迭代后训练矢量基于硬决定分配给聚类的集。在从模糊状态向硬状态转变的后期，平均最小失真度的减少变得较慢。平均最小失真度定义为：

$$D=\frac{1}{M}\sum_{i=1}^{M}d_{\min}(\boldsymbol{x}_i)=\frac{1}{M}\sum_{i=1}^{M}\min_{\boldsymbol{y}_i\in\boldsymbol{y}}d(\boldsymbol{x}_i,\boldsymbol{y}_j)\tag{3}$$

在这期间，如果 $\chi-C^{(v)}\neq 0$，则每次迭代后总失真均会减少。当减少率在 v^* 次迭代后低于门限值 ε' 时，就说明所有训练矢量均转变成硬模式，即 $C^{(v^*)}=\chi$。在随后的迭代过程中训练矢量空间就基于最近邻域条件来分割，最近邻域条件定义为：

$$\mu_j=\begin{cases}1 & \text{if}\quad d(\boldsymbol{x}_i,\boldsymbol{y}_j)=d_{\min}(\boldsymbol{x}_i)\\ 0 & \text{其它}\end{cases}\tag{4}$$

当平均最小失真减少率低于门限值 ε 时，整个码书训练过程结束。因此保证 FVQ 算法不会局部最小的条件是门限值的选取必须满足 $\varepsilon'<\varepsilon$。

文献[5]采用不同的隶属函数和码书运算公式，给出了FVQ1、FVQ2、FVQ3三种算法。FVQ1的隶属函数定义为：

$$\mu_j(\boldsymbol{x}_i)=f(d(\boldsymbol{x}_i,\boldsymbol{y}_j),d_{\max}(\boldsymbol{x}_i))=\left(1-\frac{d(\boldsymbol{x}_i,\boldsymbol{y}_j)}{d_{\max}(\boldsymbol{x}_i)}\right)^u\tag{5}$$

式中参数 u 为正整数，实验表明 $u=2$ 时，所得的码书性能较理想。FVQ1的码书运算公式与LBG算法相同，定义为：

$$\boldsymbol{y}_j=\frac{\sum_{i=1}^{M}\mu_j(\boldsymbol{x}_i)\boldsymbol{x}_i}{\sum_{i=1}^{M}\mu_j(\boldsymbol{x}_i)}\quad\forall\ j=1,2,\cdots,k\tag{6}$$

FVQ2的隶属函数和码书运算公式与FKM算法相同。隶属函数定义为：

$$\mu_j(\boldsymbol{x}_i)=\frac{1}{\sum_{l=1}^{k}\left(\frac{d(\boldsymbol{x}_i,\boldsymbol{y}_j)}{d(\boldsymbol{x}_i,\boldsymbol{y}_l)}\right)^{\frac{1}{m-1}}}\quad\forall\ j=1,2,\cdots,k\tag{7}$$

式中调节参数 $\frac{1}{m-1}$ 是正整数，实验表明 $m=1.2$ 时，训练所得码书的平均最小失真为最小。FVQ2的码书运算公式定义为：

$$\boldsymbol{y}_j=\frac{\sum_{i=1}^{M}\mu_j(\boldsymbol{x}_i)^m\boldsymbol{x}_i}{\sum_{i=1}^{M}\mu_j(\boldsymbol{x}_i)^m}\quad\forall\ j=1,2,\cdots,k\tag{8}$$

FVQ3采用FKM定义的隶属函数(7)，码书运算用LBG算法定义的运算公式(6)。FVQ1、FVQ2、FVQ3均遵循设计三要素策略，因此由上面隶属函数和码书运算公式定义可知：它们的运算量均大于LBG算法，但比FKM算法小得多。运算量由大到小的顺序是FVQ2、FVQ3、FVQ1。

3 FVQE 算法

理论上FVQ算法的推导是基于目标函数有约束的最小化。目标函数定义为：

$$J_m=\sum_{j=1}^{k}\sum_{i=1}^{M}\mu_j(\boldsymbol{x}_i)^m\|\boldsymbol{x}_i-\boldsymbol{y}_j\|^2\tag{9}$$

式中 $1\leqslant m<\infty$ 是模糊参数。m 趋于1时，训练矢量的空间分割就接近于硬决定过程；m 增大，则隶属退化，聚类模糊度随之增强。对一给定的码书矢量集，满足 $J_m=J_m(\mu_j,j=1,2,\cdots,k)$ 最小化，则要保证

$\mu_j(\boldsymbol{x}_i)\in[0,1]\forall\ i,j$，而且 $0<\sum_{i=1}^{M}\mu_j(\boldsymbol{x}_i)<M$，另

$$\sum_{j=1}^{k}\mu_j(\boldsymbol{x}_i)=1\quad\forall\ i=1,2,\cdots,M \tag{10}$$

结合前面码书设计第二个要素中隶属函数的选取条件，我们提出了指数型隶属函数，定义为：

$$\mu_j(\boldsymbol{x}_i)=\exp\left[-\frac{d(\boldsymbol{x}_i,\boldsymbol{y}_j)}{d_{\max}(\boldsymbol{x}_i)-d(\boldsymbol{x}_i,\boldsymbol{y}_j)}\right] \tag{11}$$

$$\mu_j(\boldsymbol{x}_i)=1-\exp\left[1-\frac{d_{\max}(\boldsymbol{x}_i)}{d(\boldsymbol{x}_i,\boldsymbol{y}_j)}\right] \tag{12}$$

$$\mu_j(\boldsymbol{x}_i)=\exp\left[-\tan\frac{2}{\pi}\left(1-\frac{d(\boldsymbol{x}_i,\boldsymbol{y}_j)}{d_{\max}(\boldsymbol{x}_i)}\right)\right] \tag{13}$$

选取这 3 个指数型隶属函数的主要原因是它们简单和收敛快。按照设计三要素策略进行码书矢量训练，矢量分配在量化器设计的后期是基于硬决定的，所以对 FVQ1 型量化器，可通过目标函数(9)所定义的最小化来保证。如码书计算用公式(6)，对应隶属函数(11)、(12)、(13)式，就得到 3 个模糊矢量量化算法，简称为 FVQE11、FVQE12、FVQE13。对 FVQ2 型量化器，设计后期同样是基于硬决定的，则隶属函数取值为 0 或 1，这样不论取 m 值多少，有 $\mu_j(\boldsymbol{x}_i)^m=\mu_j(\boldsymbol{x}_i)$，可满足目标函数(9)最小。码书计算用公式(8)，就得到了 FVQE21、FVQE22、FVQE23 算法。

具体算法如下：

(1)选择 ε、ε' ($\varepsilon'>\varepsilon$)和随机初始码书 $\boldsymbol{y}=\{\boldsymbol{y}_j,j=1,2,\cdots,k\}$；令 $C=0,I_i=\boldsymbol{y}\forall\ i=1,2,\cdots,M$；利用式(3)计算 D。

(2) 置。$D_{old}=D,i=0$；

(3) 令 $i=i+1$；若 $\boldsymbol{x}_i\in C$，用式(4)计算 $\mu_j(\boldsymbol{x}_i)$，到步骤

(4)；否则如果 $\boldsymbol{x}_i\notin C$ 且 $N(I_i)\geqslant2$，若 $\boldsymbol{y}_j\notin I_i$，$\mu_j(\boldsymbol{x}_i)=0$ 则；否则不同的量化器采用不同隶属函数公式计算 $\mu_j(\boldsymbol{x}_i)$，FVQE11、FVQE21 用公式(11)；FVQE12、FVQE22 用公式(12)；FVQE13、FVQE23 用公式(13)。根据式(1)计算 $d_{ave}(\boldsymbol{x}_i)$；用式(2)更新集 I_i；到步骤(4)。如果 $\boldsymbol{x}_i\notin C$ 且 $N(I_i)<2$，$C\leftarrow C\cup\{\boldsymbol{x}_i\}$；则，用式(4)计算 $\mu_j(\boldsymbol{x}_i)$。

(4)若 $i<M$，返回到(3)；否则根据公式(6)计算码书(FVQE11、FVQE12、FVQE13 算法)，根据公式(8)计算码书(FVQE21、FVQE22、FVQE23 算法)；用式(3)计算 D；如果 $C=\chi$，到步骤(5)；若$(D-D_{old})/D_{old}>\varepsilon$，返回到(2)。

(5) 若$(D-D_{old})/D_{old}>\varepsilon'$，返回到(2)；否则停止迭代。

至此我们定义了 6 个指数型模糊矢量量化算法。为了验证它们的有效性，下面给出了它们码书设计的统计性能，并和其它算法进行了比较。

4　实验结果

本节针对标准 8bits 灰度图象 Lenna 进行处理，其大小为 256×256，采用 16 维的矢量，即有 4 096个矢量作为学习矢量；码书大小为 128。表 1 列出了 LBG、FKM、FVQ、FVQE11、FVQE12、FVQE13、FVQE21、FVQE22、FVQ23 算法的统计性能。表中 ν 为收敛时所需迭代次数，D^{ν} 为次迭代后平均最小失真，$PSNR$ 是，是峰值信噪比，定义为：

$$PSNR=10\log_{10}\frac{255^2}{\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}(F_{ij}-\hat{F}_{ij})} \tag{14}$$

其中 F_{ij}和 $\hat{F}_{ij}$分别是原图象和译码图象中的象素，$N\times N$ 是图象大小，N 等于 256。为了比较它们的运算量，所有算法均在奔腾 120 计算机上，用 VISUAL C++4.1 编译程序。

表 1　算法统计性能比较($K=128,M=4\ 096$, $\varepsilon=10^{-4},\varepsilon'=10^{-2}$)

Table 1　Statistical Performance Comparison Between Several Algorithms

算法	ν	D^{ν}	PSNR (dB)	运行时间 (分钟)
LBG[1]	24	151.81	26.29	1.30
FKM[4](m=1.2)	35	105.72	27.88	47.51
FVQ1[5](u=2)	36	105.91	27.87	6.34
FVQ2[5](m=1.2)	29	104.44	27.93	18.57
FVQ3[5](m=1.2)	26	104.17	27.95	7.50
FVQE 11	32	105.84	27.86	6.29
FVQE 12	30	107.38	27.80	5.96
FVQE 13	35	105.62	27.90	6.31
FVQE 21(m=1.2)	33	105.53	27.85	18.38
FVQE 22(m=1.2)	37	105.94	27.83	18.56
FVQE 23(m=1.2)	31	104.44	27.92	17.00

我们对每种算法进行了 5 次仿真，将其统计结果列于表 1 中。由表 1 可见 6 种 FVQE 算法的编码性能跟 FVQ 算法接近，均比 FKM 算法快，比 LBG 性能好。而且 FVQE1 型算法比 FVQ1 算法快，FVQE2 型算法略快于文献[5]中的 FVQ2 算法。同时发现，随机选取的码书对 FVQE 设计结果影响很小。

5 小　结

本文分析了模糊矢量量化算法的原理，提出了 6 种指数型模糊矢量量化算法，算法具有收敛快、简单等优点。实验结果也是令人满意的。FVQE 的图象编码性能与 FVQ 相当，但收敛速度要略快于 FVQ 算法。本文算法也可用于其它随机数据的聚类处理，但如何进一步提高算法的收敛速度，即快速算法的研究，将是我们下一步探讨的课题。

参考文献

1 Gray R M. Vector quantization. IEEE ASSP Magazine, 1984,1: 4～29.

2 Zeger K, et al. Globally optimal vector quantizer design by stochastic relaxation. IEEE Trans. on Signal Processing, 1992, 40(2):310～312.

3 Yair E,et al. Competitive learning and soft competition for vector quantizer design. IEEE Trans. on Signal Processing, 1992, 40(2):294～309.

4 Bezdek J C,et al. FCM: The fuzzy c-mean clustering algorithm. Comput. Geosciences, 1984,10(2～3):191～203.

5 Karayannis N,et al. Fuzzy vector quantization algorithm and their application in image compression. IEEE Trans. on Image Processing, 1995,4(9):1193～1201.

张基宏，男，1992 年 3 月毕业于东南大学无线电系，博士，深圳大学电子系副教授。1995 年底由英国中英文化交流协会科研项目资助，在英国 Lancashire 大学进行合作搞科研。现在日本创价大学工学部从事数字图象处理和编码等方向的研究。先后发表论文 10 余篇。

王晖，男，讲师，1969 年生，1996 年在西安交通大学获工学博士学位，主要研究方向为信息处理，图象处理，模式识别，发表论文 10 余篇。

Yoshito Ueno，63 岁，教授，1979 年获日本早稻田大学博士学位，1993 年在英国 Strathclyde 大学任客座教授，研究方向为智能化信息处理，已发表论文 40 余篇。

Fuzzy Vector Quantization for Image Coding

Zhang Jihong, Wang hui

(*Department of Electronics, Shenzhen University, Shenzhen* 518060)

Yoshiti Ueno

(*Department of Information System Science, Faculty of Engineering, Soka University, Tokyo* 192)

Abstract This paper analyses the principle of fuzzy vector quantization (FVQ) for image coding, gives three strategies of FVQ algorithm during the codebook design process, and presents six new fuzzy vector quantization algorithms with exponential membership function for image coding, named FVQE algorithm. Simulation results show that the performance of FVQE is comparable to FVQ for image compression, and a slight improvement on the convergence speed is obtained.

Keywords Image coding, Fuzzy vector quantization, Fuzzy K-means algorithm