

文章编号:1004-9037(2012)06-0639-07

最大熵矢量量化及其在TMS320DM642上的实现

汪全全^{1,2} 王靖琰^{3,4,5} 李勇平^{1,2}

- (1. 中国科学院上海应用物理研究所, 上海, 201800; 2. 复旦大学上海市智能信息处理重点实验室, 上海, 200433;
3. 阿卜杜拉国王科技大学计算机、电气与数学科学工程系, 图瓦, 沙特阿拉伯, 23955-6900;
4. 南京大学计算机软件新技术国家重点实验室, 南京, 210093;
5. 湖南师范大学高性能计算与随机信息处理教育部重点实验室, 长沙, 410012)

摘要:为了克服传统的矢量量化方法存在的信息损失量大及二次规划(Quadratic programming, QP)量化方法计算复杂度大等缺点,提出了一种新的量化方法——最大熵量化。这种量化方法一方面能将量化权值的熵最大化,从而确保在没有先验知识的情况下不会造成太多量化误差;另一方面则考虑了矢量集合在时间空间上的分布关系。本文在TMS320DM642处理器上实现了这种算法,并进行了一系列的算法和程序层的优化。在基于图像的目标识别应用中的实验证明,最大熵矢量量化算法及其在TMS320DM642上的实现,不仅能提高识别的性能,而且能满足实时性的要求。

关键词:矢量量化;最大熵;QP量化;0-1量化;目标函数

中图分类号:TP391.4

文献标识码:A

Maximum Entropy Vector Quantization and Its Implementation on TMS320DM642

Wang Quanquan^{1,2}, Wang Jingyan^{3,4,5}, Li Yongping^{1,2}

- (1. Shanghai Institute of Applied Physics, Chinese Academy of Sciences, Shanghai, 201800, China;
2. Shanghai Key Laboratory of Intelligent Information Processing, Shanghai, 200433, China;
3. Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology, Thuwal, 23955-6900, Saudi Arabia;
4. State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, 210093, China;
5. Key Laboratory of High Performance Computing and Stochastic Information Processing, Ministry of Education of China, Hunan Normal University, Changsha, 410012, China)

Abstract: To overcome the shortages of traditional vector quantization(VQ) and quadratic programming(QP) VQ algorithm, a new quantitative method i. e. the maximum entropy quantization is proposed. The quantization approach can maximize the entropy of the quantification weights, thus avoiding too much quantization error when a priori knowledge is absent. And it considers the space-time distribution of vector collection. The TMS320DM642 processors are used to implement the algorithm with a series of optimization on algorithm and program levels. Experiments on object recognition prove that the maximum entropy vector quantization algorithms with acceleration on TMS320DM642 processor can improve the recognition performance and meets the real-time requirements.

Key words: vector quantization; maximum entropy; quadratic programming(QP) quantization; 0-1 quantization; object function

基金项目:上海市智能信息处理重点实验室开放课题(HIPL-2011-003)资助项目;南京大学计算机软件新技术国家重点实验室开放课题(KFKT2012B17)资助项目;高性能计算与随机信息处理教育部重点实验室开放课题(HS201107)资助项目。

收稿日期:2012-05-07;**修订日期:**2012-10-09

引 言

矢量量化(Vector quantization, VQ)是对矢量进行量化,把矢量空间分成若干个小区域,在每个小区域中寻找一个代表矢量,量化时落入某小区域的矢量就用该区域的代表矢量表示。矢量量化技术是一种数据压缩和编码技术,几乎能涵盖各领域图像、视频、语音等的压缩、存储、传输、编码与识别等应用。矢量量化技术的关键在于码本的建立和码字搜索算法。关于码本的生成,Linde, Buzo 和 Gray 最先提出了 LBG 算法^[1],该算法从一组码矢量出发,进而对所有图像矢量进行划分,然后再重新计算码矢量,直到码矢量的变化收敛。Constantinescu 和 Storer 在此基础上提出了自适应矢量量化方法,使用变字长的图像块(矢量)和码字,自适应地更新码本^[2-3]。在自适应矢量量化方法的基础上,Narabadi, Lee 等提出了基于 Kohonen 神经网络的自组织特征映射(Self-organizing feature mapping, SOFM)矢量量化技术^[4-5],它通过领域作用使获胜神经元拓扑领域内的各码字都得到更新;Ahalt 等提出了基于竞争学习(Competing learning, CL)的矢量量化方法^[6]。国内有学者对 SOFM 及 CL 提出了改进型研究方法,以解决基于 Kohonen SOFM 算法上存在的码字利用不均匀的问题^[7-8]。为了提高码字利用率,减小运算量,提高信源匹配度,也有不少人提出新的初始码本设计算法^[9]。但是传统的矢量量化方法简单地将一个矢量量化到码本中最近邻的一个码字中,使得此种以“0-1”离散化方式对现实世界中具有连续性的矢量进行量化时造成不可避免的信息损失。Wang Jingyan 等突破了这种量化思路,创造性地提出了将一个矢量量化到多个码字的方法(多重量化),并突破“0-1”量化的思路,提出连续实数的量化权值的二次规划计算方法,称为 QP 量化^[10]。但是这种算法基于 QP 问题,计算复杂度较大,且无法并行化,故难以适应嵌入式系统上图像、语音压缩和识别的应用要求,同时它还忽略了不同矢量之间在时间空间上的相邻关系,也造成了一定程度的信息损失。因此,本文提出了基于熵最大化和时间空间紧邻的量化方法以克服由 QP 量化造成的信息损失。该方法将矢量量化的量化权值的熵进行最大化计算,同时保留时间空间上的紧邻关系,另外,由于算法对不同权值的计算是独立的,也就为并行化运算创造了条件。

本文在 TMS320DM642 处理器上实现了最大熵矢量量化算法,并进行一系列算法和程序层的优化,在基于图像的目标识别和语音口令识别的应用中进行实验验证,该算法及其在 TMS320DM642 上的实现不仅能提高识别的性能,还能满足实时性的要求。

1 算法详细介绍

1.1 矢量量化算法

给出一组矢量的集合 $\chi = \{x_1, \dots, x_n\}$, 共有 N 个矢量。这些矢量可以是一幅图像的局部特征的集合,也可以是一段语音信号的各帧特征的集合。另外给出一个码本 $\nu = \{v_1, \dots, v_k\}$, 共包含 k 个码字。矢量量化示意图如图 1 所示。

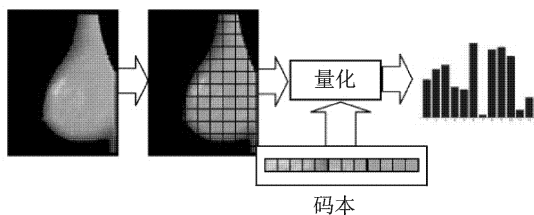


图 1 矢量量化示意图

矢量量化算法的任务是:将 χ 中的所有矢量量化到 ν 中的码字上去,用码字来表达矢量。进而统计量化到每个码字的矢量的数量,得到直方图,作为这个矢量集合 χ 的特征表达。

为了表达这一过程,为每个 $x_i \in \chi$ 定义一个量化权值矢量 $h_i = [h_{i1}, \dots, h_{ik}]^T$, 其中 h_{ki} 为第 i 个矢量量化到第 k 个码字的权值。而最终量化直方图 $f = [f_1, \dots, f_k]^T$ 则是通过对所有 i 的量化权值矢量进行累加得到

$$f_k = \sum_{i=1}^N h_{ki} \quad (1)$$

对 h_{ki} 的计算有不同的方法,从而引出不同的量化策略:

(1) 0-1 量化 在传统的矢量量化中,一般先在码本 ν 中寻找某个 $x_i \in \chi$ 的最近邻码字 ν_k^* , 进而将其量化到此码字。具体有

$$k^* = \operatorname{argmin} \|x_i - v_k\|^2 \quad (2)$$

进而得到其量化权值矢量

$$h_{ki} = \begin{cases} 1 & k = k^* \\ 0 & \text{其他} \end{cases} \quad (3)$$

由于量化值非 0 即 1,称之为 0-1 量化。

(2) QP 量化 由于 0-1 量化在量化过程中,强

行将具有连续特征的矢量量化到离散的码本中,造成了信息的损失。为此,在先前的工作中提出了软量化的概念,这一概念有以下两个特点:(1)突破传统量化方法中只能将 \mathbf{x}_i 量化到码本中的某一个码字 \mathbf{v}_k^* 的局限,而是将其量化到某若干个码字上;(2)突破传统量化方法中量化权值非1即0的限制,将量化权值扩展到一个连续值。

为此,先从码本中寻找距离 \mathbf{x}_i 最近的 $|N_i|$ 个码字,组成其实际量化的字码本 N_i ;然后,提出以下假设:通过这个子码本中的码字 $\mathbf{v}_k \in N_i$,可以通过加权平均的方法重构出 \mathbf{x}_i ,其权值即是量化权值

$$\mathbf{x}_i = \sum_{\mathbf{v}_k \in N_i} h_{ki} \mathbf{v}_k \tag{4}$$

基于以上假设,可以通过最小化重构误差的平方欧拉距离来得到

$$\begin{aligned} \min_{h_{ki}} \quad & \left\| \mathbf{x}_i - \sum_{\mathbf{v}_k \in N_i} h_{ki} \mathbf{v}_k \right\|^2 = h_{ki} g_{kl} h_{li} \\ \text{s. t.} \quad & \sum_{k \in N_i} h_{ki} = 1, h_{ki} \geqslant 0 \end{aligned} \tag{5}$$

式中 $g_{kl} = (\mathbf{x}_i - \mathbf{v}_k)^\text{T} (\mathbf{x}_i - \mathbf{v}_l)$ 。

这一优化问题可以通过标准的二次规划(QP)方法解决,所以称之为QP量化。该方法在2011年提出,并首次应用于医学图像的表达和检索^[10]。

1.2 熵最大化矢量量化

QP量化虽然得出了迄今为止最优的量化效果,但是同时也存在着严重的缺陷。这种算法基于二次规划,且是计算复杂度较大。为此,可以通过分布式计算以及多核并行等策略来加速计算。但是这些策略仅限于台式计算机以及工作站等设备。

随着嵌入式系统的发展,以及图像压缩,语音识别等基于矢量量化的应用在手持设备上的普及,需要能够在运算和存储能力都非常有限的嵌入式设备上能高效实现的量化算法。为此,需开发一种既能对矢量进行有效的量化,同时又能在嵌入式设备中运行的算法。

1.2.1 算法设计目标

量化权值的计算方法,其权值应满足以下条件
(1)反映矢量和码字之间的相似和差异关系。矢量到相似的码字的量化权值应较大,反之亦然。

(2)在没有先验知识的情况下,量化权值分布应尽量平坦。从信息论的角度,量化权值对应的熵应该最大化。在时间、空间上具有相邻关系的码字,在量化值方面也应该具有相似性,以反映原始矢量集合的时空分布特点。

(3)计算简单,可并行运算。为实现该算法的设

计,采用先定义目标函数,然后进行数学优化,最后进行算法设计。

1.2.2 目标函数

目标函数的设计对应以上提出的各个设计目标,每个设计目标将设计出对应的目标函数项。

(1)为了能反映矢量和码字之间的相似和差异性关系,如果矢量 \mathbf{x}_i 到码字 \mathbf{v}_k 的平方欧拉距离 $d_{ki} = \|\mathbf{x}_i - \mathbf{v}_k\|^2$ 较大,则量化权值 h_{ki} 应较小;反之亦然。因此,求取的 h_{ki} 应使以下函数最小化

$$O^1(h_{ki}) = \sum_{i=1}^N \sum_{k=1}^K d_{ki} h_{ki} \tag{6}$$

(2)从信息论角度, h_{ki} 可以看做是 \mathbf{x}_i 量化到 \mathbf{v}_k 的概率。对于量化系统,其对应的熵为

$$- \sum_{i=1}^N \sum_{k=1}^K h_{ki} \log(h_{ki}) \tag{7}$$

为了将此熵函数最大化,需将以下函数最小化

$$O^2(h_{ki}) = \sum_{i=1}^N \sum_{k=1}^K h_{ki} \log(h_{ki}) \tag{8}$$

另外,由于 h_{ki} 被视作为是概率分布,则应服从概率分布的限制条件

$$\sum_{i=1}^N \sum_{k=1}^K h_{ki} = 1, h_{ki} \geqslant 0 \tag{9}$$

(3)为了表示不同矢量在时间、空间上的相邻关系,定义一个相邻指示矩阵 $\mathbf{W} = [\mathbf{w}_{ij}]^{N \times N}$,其中

$$\mathbf{w}_{ij} = \begin{cases} 1 & \mathbf{x}_i \text{ 和 } \mathbf{x}_j \text{ 相邻} \\ 0 & \text{其他} \end{cases} \tag{10}$$

如图2所示,对一段语音信号,通过分帧,得到 $\{\mathbf{x}_1, \cdots, \mathbf{x}_5\}$ 5个矢量。其中与 \mathbf{x}_3 相邻的有 \mathbf{x}_2 和 \mathbf{x}_4 两帧,则对应的相邻指示矩阵为

$$\mathbf{W} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{11}$$

基于这个相邻指示矩阵,对于两个矢量 \mathbf{x}_i 和 \mathbf{x}_j ,如果它们相邻,则它们对应的量化权值矢量 \mathbf{h}_i 和 \mathbf{h}_j 之间的欧拉距离的平方 $\|\mathbf{h}_i - \mathbf{h}_j\|^2 = \sum_{k=1}^K (h_{ki} - h_{kj})^2$ 应该较小。因此,定义如下目标函数

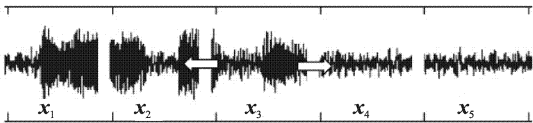


图2 相邻的语音矢量示意图

$$\begin{aligned}
O^3(h_{ki}) &= \sum_{i,j=1}^N \omega_{ij} \| \mathbf{h}_i - \mathbf{h}_j \|^2 = \\
&\sum_{i,j=1}^N \omega_{ij} (\mathbf{h}_i - \mathbf{h}_j)^T (\mathbf{h}_i - \mathbf{h}_j) = \\
&\sum_{i,j=1}^N \omega_{ij} (\mathbf{h}_i^T \mathbf{h}_i - 2\mathbf{h}_i^T \mathbf{h}_j + \mathbf{h}_j^T \mathbf{h}_j) = \\
&2 \sum_{i=1}^N \left(\sum_{j=1}^N \omega_{ij} \right) (\mathbf{h}_i^T \mathbf{h}_i) - 2 \sum_{i,j=1}^N \omega_{ij} (\mathbf{h}_i^T \mathbf{h}_j) = \\
&2 \left(\sum_{i=1}^N u_i \sum_{k=1}^K h_{ki}^2 - \sum_{i,j=1}^N \omega_{ij} \sum_{k=1}^K h_{ki} h_{kj} \right) \quad (12)
\end{aligned}$$

式中 $u_i = \sum_{j=1}^N \omega_{ij}$ 。

通过最小化以上目标函数,可以达到量化权值矢量保持原始图像或者语音信号的时间、空间紧邻关系的目的。

最终的目标函数是以上 3 个目标函数项的加权和

$$O(h_{ki}) = O^1(h_{ki}) + \alpha O^2(h_{ki}) + \beta O^3(h_{ki}) \quad (13)$$

式中: α 和 β 是各项子函数的权值。以量化权值为变量,优化以上目标函数,就可以得到同时满足 1.2.1 节中提出的量化目标。

1.2.3 数学优化

根据设计的目标函数,量化问题转化为以下优化问题,目的是求取量化权值 $\{h_{ki}\}, k=1, \dots, K, i=1, \dots, N$, 以使得目标函数最小

$$\begin{aligned}
\min_{h_{ki}} O(h_{ki}) &= O^1(h_{ki}) + \alpha O^2(h_{ki}) + \beta O^3(h_{ki}) = \\
&\sum_{i=1}^N \sum_{k=1}^K d_{ki} h_{ki} + \alpha \sum_{i=1}^N \sum_{k=1}^K h_{ki} \log(h_{ki}) + \\
&2\beta \left(\sum_{i=1}^N u_i \sum_{k=1}^K h_{ki}^2 - \sum_{i,j=1}^N \omega_{ij} \sum_{k=1}^K h_{ki} h_{kj} \right) \\
&\text{s. t. } \sum_{i=1}^N \sum_{k=1}^K h_{ki} = 1, h_{ki} \geq 0 \quad (14)
\end{aligned}$$

可以证明,目标函数的最小值不可能取在 $h_{ki}=0$ 时,原因在于当 $h_{ki}=0$ 时,目标函数中的 $h_{ki} \log h_{ki}$ 项的导数为负无穷。所以在优化时,无需担心 $h_{ki} \geq 0$ 无法满足,故可忽略此约束项。为了利用拉格朗日乘子法对目标函数进行优化,可以将式(14)中的优化问题改为

$$\begin{aligned}
\min_{h_{ki}} O(h_{ki}) \\
\text{s. t. } g(h_{ki}) = 0 \quad (15)
\end{aligned}$$

其中 $g(h_{ki}) = \sum_{i=1}^N \sum_{k=1}^K h_{ki} - 1$ 为约束条件。此优化问题对应的拉格朗日函数为

$$L = O(h_{ki}) + \lambda g(h_{ki}) =$$

$$\begin{aligned}
&\sum_{i=1}^N \sum_{k=1}^K d_{ki} h_{ki} + \alpha \sum_{i=1}^N \sum_{k=1}^K h_{ki} \log(h_{ki}) + \\
&2\beta \left(\sum_{i=1}^N u_i \sum_{k=1}^K h_{ki}^2 - \sum_{i,j=1}^N \omega_{ij} \sum_{k=1}^K h_{ki} h_{kj} \right) + \\
&\lambda \left(\sum_{i=1}^N \sum_{k=1}^K h_{ki} - 1 \right) \quad (16)
\end{aligned}$$

式中: λ 是 $g(h_{ki})=0$ 约束项对应的拉格朗日乘数。

对 L 求 h_{ki} 的偏导数并置零得到

$$\frac{\partial L}{\partial h_{ki}} = d_{ki} + \alpha(\log(h_{ki}) + 1) + 2\beta(2u_i h_{ki} - \sum_{j=1}^N \omega_{ij} h_{kj}) + \lambda = 0 \quad (17)$$

可以证明,最优解 $\{h_{ki}^*, \lambda^*\}$ 应满足以上等式。

1.2.4 算法设计

直接对式(17)求闭合解是困难的,故本文采用迭代算法的思想。根据式(18),逐次地更新 $\{h_{ki}\}, \lambda$ 变量,使其逼近最优解。为了对式(17)进行优化,对式(17)稍作变换得到

$$\begin{aligned}
h_{ki} &= \exp \left[\frac{-d_{ki} - 2\beta \left(2u_i h_{ki} - \sum_{j=1}^N \omega_{ij} h_{kj} \right)}{\alpha} \right] \cdot \\
&\exp \left[-\frac{\lambda}{\alpha} - 1 \right] = \\
&\exp \left[\frac{-d_{ki} - 2\beta \left(2u_i h_{ki} - \sum_{j=1}^N \omega_{ij} h_{kj} \right)}{\alpha} \right] \cdot \gamma \quad (18)
\end{aligned}$$

式中 $\gamma = \exp \left[-\frac{\lambda}{\alpha} - 1 \right]$ 是跟 h_{ki} 无关的常数。可以证明,最优的 h_{ki}^* 满足以上等式。为了满足概率分布的限制条件(9),对 h_{ki} 进行归一化操作 $h_{ki} = \frac{h_{ki}}{\sum_{i=1}^N \sum_{k=1}^K h_{ki}}$, 从而约掉了 γ 项,所以在求解 h_{ki} 时可以将其忽略。

第 t 次迭代中,重复以下两个步骤:

(1) 利用上次更新得到的 $\{h_{ki}^{t-1}\}$, 代入等式(18)中右边项,从而更新左边项中的 h_{ki} 为 h_{ki}^t

$$h_{ki}^t = \exp \left[\frac{-d_{ki} - 2\beta \left(2u_i h_{ki}^{t-1} - \sum_{j=1}^N \omega_{ij} h_{kj}^{t-1} \right)}{\alpha} \right] \quad (19)$$

(2) 对 $\{h_{ki}^{t-1}\}$ 进行归一化操作 $h_{ki} = \frac{h_{ki}}{\sum_{i=1}^N \sum_{k=1}^K h_{ki}}$ 。直到收敛或者达到最大迭代次数 T 。为了对 h_{ki} 进行初始化,定义 h_{ki} 的初始值为

$$h_{ki}^0 = \exp \left(-\frac{d_{ki}}{\alpha} \right) \quad (20)$$

2 算法在 DSP 上的实现

为了验证提出的量化算法在图像表达方面的识别性能以及其在嵌入式平台上的实时性,选择瑞泰公司提供的 ICETEK-DM642 评估板做测试平台。该评估板采用 TI 公司的 TMS320DM642 视频图像定点 DSP 处理器。此系统是根据采集的图像识别其图像内容。在实验中,采集了 5 种目标物体的图像作为数据库。其图像大小为 256 像素×256 像素。如图 3 所示。

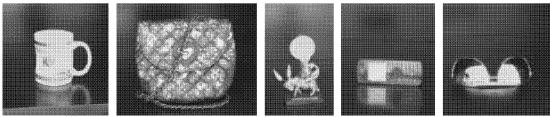


图 3 目标物体图像示例

对于采集的图像,首先确定目标区域,然后将目标区域划分为多个 16×16 的小图像块,作为图像的局部特征,也就是矢量量化中的矢量集合。每幅图像共有 256 个小图像块。然后即采用最大熵矢量量化算法得到目标图像的直方图作为这个目标区域的图像特征,最后采用最近邻方法从数据库中搜索同类图像,得到属于那个目标的判决结果。如图 4 所示。其中矢量量化中用到的码本,通过对数据库中的图像的局部特征集合进行聚类得到,并实现以数组的形式存储在内存中。采用的是 PAL 制式的输入图像。

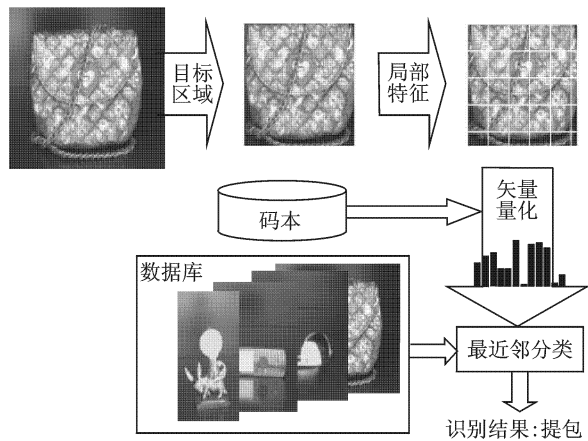


图 4 基于矢量量化的图像目标识别流程图

3 算法在 DSP 上的优化

为了实现量化算法在 TMS320DM642 芯片上的加速,采用了以下程序优化技术:

(1)使用dsplib 和imglib:这两个库都是由汇编

语言直接写成,具有极高的处理能力。例如在式(19)中,需要计算 $\sum_{j=1}^N w_{ij} h_{kj}^{t-1}$,可以调用 dsplib 所提供的函数 int DSP_dotprod(const short * restrict x, const short * restrict y, int nx)进行高效的计算。

(2)设置编译选项:为了提高软件流水线质量,增加程序并行程度,修改了编译器的设置,选中了如下选项:-o3:最高程度优化,编译器将执行各种优化循环方法,如软件流水,循环展开;-pm:程序级优化,使优化器访问整个程序;-op3:使得编译器可以使用其他文件中定义的函数。

(3)For 循环展开:在执行式(18,19)中的计算时,很明显,不同的 h_{ki}^t 的计算之间是独立的,这就为将循环展开,从而将程序并行化提供了可能。将多次循环集中在一个循环中同时执行,而不是等上次循环执行完毕后才开始下一次,从而增加了算法的并行程度。如图 5 所示。

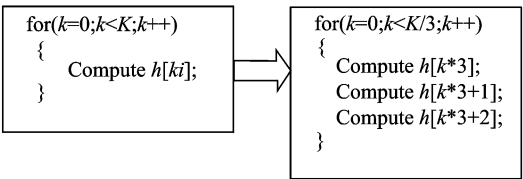


图 5 For 循环展开示意图

(4)使用内联函数:C64+编译器提供的 intrinsic 函数可以快速优化C 代码,intrinsic 用前下划线表示(_),使用了很多内联函数来进行优化,比如内联函数

```
int mpy (int src1, int src2)
```

这一函数可以实现快速的乘法功能,但是不会进行调用函数的操作。

4 实现的性能与结果

为了对算法进行评估,从两个方面进行了测试:识别率和实时性。

4.1 识别性能测试

(1)自建数据库测试:对于图像目标识别,采集了 30 幅图像作为测试样本进行识别实验,每种目标 6 幅图像,作为测试集合。另外将最大熵量化方法跟传统的 0-1 量化和 QP 量化做了对比,实验结果如表 1 所示。

很明显,本文提出的最大熵量化算法在识别率方面超越了前两种方法。

表 1 识别率比较结果 %

试验 内容	识别率		
	0-1 量化	QP 量化	最大熵量化
自建数据库	60.00	76.67	93.33
公共数据库	62.15	71.39	87.29

(2)公共数据库测试:另外,为了表示公平的比较,也在公开的物体识别数据库 Columbia image object library (COIL20)上做了识别率的比较。这个数据库包含了 20 种物品的 1 440 幅图像,每种物品 74 幅图像。采用留一法进行测试,即每次用一幅图像作为测试集,而其他图像作为训练集。共重复 1 440 次,对每次的测试结果进行平均得到最终的识别结果。实验结果如表 1 所示。

很明显,本文提出的最大熵量化算法在识别率方面超越了其他两种方法。主要原因在于:(1)本文算法考虑了矢量在时间空间上的分布;(2)本文算法考虑了量化值的熵的最大化,使得在没有先验知识的情况下,量化错误的机率减少。

4.2 实时性测试

为了测试算法在优化前后的实时性,针对每项优化方法进行了实时性测试。

(1)使用 dsplib 和 imglib:采用 C 语言编写与 DSP_dotprod 函数相同功能的 C 代码,发现采用 dsplib 和 imglib 库的函数可以大大提高并行化程度,将执行程序的时钟数从 9 602 降低到 178,仅为原来的 1.85%。

(2)设置编译选项:为了测试编译选项对程序执行效率提高的影响,统计了使用 -o3, -pm 和 -op3 优化选项后对一帧图像进行识别所需要的时钟数,如图 6 所示。由图可见,使用 -o3 选项后,执行效率有较明显的提升,时钟数从 17 346 降低到了 15 622,为原来的约 90%。而再适用另两个选项后,变化并不明显。

(3)For 循环展开:为了测试 For 循环展开对程序效率提高的影响,对一个执行 256 次的 For 循环进行了展开,并测试了时钟数。

由图 7 可见,4 阶循环展开后,时钟数由 270 409

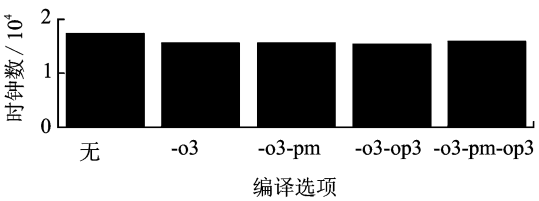


图 6 不同编译选项生成的代码执行时钟数比较

降为 34 810,是原来的 12.8%,而进行 8 阶展开后,则将为 22 872,为原来的 8.4%,都得到了大幅的执行效率上的提升,同时也使得程序代码量增加。

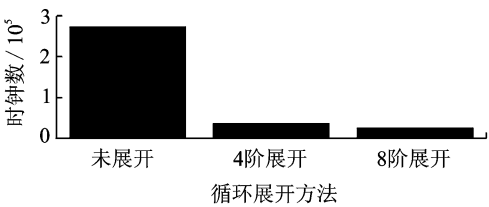


图 7 循环展开的代码执行时钟数比较

(4)使用内联函数:为了测试内联函数对程序执行效率的提升,比较了算法中大量使用的乘法功能的实现,对内联函数_mpy 做乘法和使用 * 指令做乘法进行了比较。使用了内联函数之后,可以得到同样的结果,但是一次乘法的时钟数从 22 降到了 10,为原来的 45.45%,这一操作在程序中广泛使用,大大提高了效率。

5 结束语

本文中提出了基于熵最大化和时间空间近邻关系的量化方法。这种方法能将矢量量化的量化权值的熵进行最大化,同时保留时间和空间上的近邻关系;另外,本文算法对不同权值的计算是独立的,为并行化创造了条件。本文在 TMS320DM642 处理器实现了这种算法,并进行一系列的算法和程序层的优化。在基于图像的物体识别中的实验证明,算法及其在 TMS320DM642 处理器上的实现与优化,不仅能提高识别的性能,还能满足实时性的要求。

参考文献:

[1] Linde Y, Buzo A, Gray R M. An algorithm for vector quantization design[J]. IEEE Trans on Communication, 1980, COM-28: 84-95.

[2] Constantinescu C, Storer J A. Online adaptive vector quantization with variable size codebook entries [J]. Information Processing and Management, 1994, 30(6), 745-758.

[3] Constantinescu C, Storer J A. Improved techniques for single-pass adaptive vector quantization [C] // Proceedings of the IEEE. Baltimore, MD, USA: IEEE, 1994b, 82(6), 933-939.

[4] Nasrabadi N M, Feng Y. Vector quantization of images based upon the Kohonen self-organizing feature maps [C] // Proceedings of IEEE Int’l Conference on Neural Networks. San Diego, California, USA: IEEE, 1988: 101-108.

[5] Lee T, Peterson A M. Adaptive vector quantization using a self-development neural network[J]. IEEE Journal of Select Areas in Communication, 1990, 8(8): 1458-1471.

[6] Ahalt S C, Krishnamarthy A K, Melton D E, et al. Competitive learning algorithms for vector quantization [J]. Neural Networks, 1990, 3: 277-290.

[7] 段勇,徐心和,崔宝侠. 改进的SOFM 及其在矢量量化中的应用[J]. 系统仿真学报,2006, 18(3):718-721.

Duan Yong, Xu Xinghe, Cui Baoxia. Improved SOFM and its application on vector quantization[J]. Journal of System Simulation, 2006,18(3):718-721.

[8] 马勇,阮洋. 基于改进的SOFM 神经网络的矢量量化方法[J]. 计算机工程与科学,2011,33(12):126-129.

Ma Yong, Ruan Yang. Vector quantization method based on improved SOFM neural networks[J]. Computer Engineering and Science, 2011, 33(12): 126-129.

[9] 陈善学. 矢量量化的初始码书算法[J]. 计算机工程与应用,2010, 46(1):26-28.

Chen Shanxue. Initial code book algorithm of vector quantization[J]. Computer Engineering and Science, 2010,46(1):26-28.

[10] Wang Jingyan; Almasri I. Modeling multiple visual words assignment for bag-of-features based medical image retrieval[C]// Proceedings of the IASTED International Conference on Computer Graphics and Imaging, CGIM 2012. Grete, Greece:ACTA Press, 2012: 217-224.

作者简介:汪全全(1987-),男,博士研究生,研究方向:模式识别,反应堆控制,E-mail:wangquanquan@sinap.ac.cn;王靖琰(1985-),男,博士,研究方向:机器学习,数据挖掘,生物信息学,生物特征识别,医学成像;李勇平(1963-),男,高级研究员,博士生导师,研究方向:嵌入式系统,图像处理,模式识别,机器学习,计算机视觉,反应堆控制与测量。