

ソートされたコードブックを用いた 高速ベクトル量子化

5K-1

中野恵一^{1,2} 笠原博徳²

オリンパス光学工業(株)¹ 早稲田大学理工学部²

1. はじめに

ベクトル量子化は能率の良いデータ圧縮法として、音声や画像の量子化に利用されている。このベクトル量子化は、コードブック $\{Y_1, Y_2, \dots, Y_N\}$ から入力ベクトル $X=(X_1, X_2, \dots, X_K)$ との歪みが最も小さい出力ベクトル $Y_i=(Y_{i,1}, Y_{i,2}, \dots, Y_{i,K})$ を探索し、 X を Y_i に写像することと考えられる。ここで K はベクトルの次元数、 N はコードブックのサイズである。

このベクトル量子化を実現する基本的な手法として全探索法[1]が知られている。これは、入力されたベクトルに対する最適出力ベクトルを発見するために、すべてのコードブックベクトルとの歪みを計算するもので、量子化時間が大きい。そこで、1)木探索ベクトル量子化[2]のように構造的なコードブックを用いる手法や、2)全探索法と同じコードブックを用いながら歪みの計算回数を減らすことで高速化する手法[3]などの、様々な高速化手法が提案されている。

構造的なコードブックを用いれば計算量は大幅に削減されるが、量子化歪みは全探索法に比べて一般に増大する。そこで本稿では、量子化歪みを増大させることなく、高速にベクトル量子化を実現する、ソートされたコードブックを用いた探索手法を提案する。この手法は、探索が終了した範囲での最小歪みである「歪みの上限値」と、最適出力ベクトルとの歪みとその値より小さくなることはないと保証された「歪みの下限値」との比較により、探索領域を限定することで、歪み計算回数を減少させるものである。ここで「歪みの下限値」はソートされたコードブックを用いて求められる。

なお以下では、歪みとしては一般に用いられることの多い2乗歪みを対象とする。ただし提案する手法では、入力ベクトルとコードブックベクトルとの歪みを、ベクトルの次元ごとにそれぞれ計算しそれらすべての次元にわたる和を量子化歪みとするものであれば、他の歪み測度を用いることも可能である。

2. 「歪みの部分和」を用いた手法

ここではBei and Gray[3]により提案された、「歪みの部分和」を用いた高速化手法について説明する。これは「歪みの部分和」と「歪みの上限値」とを比較し、さらにより「歪みの上限値」を与えない歪み計算は途中で打ち切ることで、歪み計算回数を減少させる手法である。ここで「歪みの部分和」 d_M とは、

$$d_M = \sum_{j=1}^M (X_j - Y_{i,j})^2, \quad 1 \leq M \leq K$$

で表される。また「歪みの上限値」とは探索が終了

した範囲での最小歪みであり、

$$d_{upper} = \min_i \left\{ \sum_{j=1}^K (X_j - Y_{i,j})^2 \right\}$$

($Y_i \in$ 探索が終了したコードブックベクトル)

と定義される。このことから本手法によれば、歪みの計算過程で「歪みの部分和」 d_M が「歪みの上限値」 d_{upper} より小さくならない($d_M \geq d_{upper}$)と判定されたならば、その次元以降の歪み計算は無駄であるから、省略することができる。これにより、全探索法に比べ、計算量を削減することができる。

3. ソートされたコードブックを利用した手法

ここでは、「歪みの上限値」と、ソートされたコードブックを利用して求められる「歪みの下限値」の比較により探索範囲を限定することで、最適出力ベクトルを高速に探索する手法を提案する。このソートされたコードブックは、次の手順で探索前に求められる。まずコードブック $\{Y_1, Y_2, \dots, Y_N\}$ を構成する各ベクトルから、同じ次元 j ($1 \leq j \leq K$) の要素を取り出し、 $W_j=(Y_{1,j}, Y_{2,j}, \dots, Y_{N,j})$ と構成されるベクトル群 $\{W_1, W_2, \dots, W_j, \dots, W_K\}$ に変換する。そしてその要素を非減少順(非増大順でも可)にソートする。このソートされたベクトル群を $\{Z_1, Z_2, \dots, Z_j, \dots, Z_K\}$ 、ただし $Z_j=(Z_{1,j}, Z_{2,j}, \dots, Z_{N,j})$ とする。また「歪みの下限値」とは、入力ベクトルとそれに対する最適出力ベクトルとの歪みとその値より小さくなることはないと保証された値であり、

$$d_{lower} = \sum_{j=1}^K d_{lower,j} = \sum_{j=1}^K \min_p (X_j - Z_{p,j})^2$$

($Z_p \in$ 探索の過程で、最適解を与えないことが判明したベクトルを除いたコードブックベクトル)

と定義される。ここで「歪みの次元ごとの最小値」 $d_{lower,j}$ は、次元ごとにソートされたベクトル Z_j から、入力ベクトルに最も近い要素を選択して得られる。この様子を8 bit画像を例に、図1を用いて説明する。

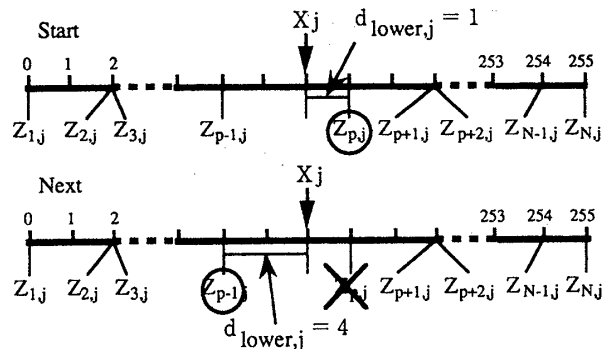


図1: 歪みの次元ごとの最小値を求めるプロセス
例えば、ある次元 j のベクトル要素をソートすると、

0 から255の範囲の数直線上に分布 ($Z_{1,j}$, $Z_{2,j}$, ..., $Z_{N,j}$) する。図中Startで示す初期状態では、数直線上で入力ベクトル要素 X_j に最も近い要素は $Z_{p,j}$ であるから、この次元 j の最小歪みは1である。次にNextで示すように、例えば $Z_{p,j}$ を含むベクトルが d_{upper} と d_M の比較により最適出力ベクトルでないと判明して除外されると、次に近い $Z_{p-1,j}$ が選択される。このとき次元 j の最小歪みは1から4へと増加する。すなわち「歪みの下限値」 d_{lower} は最適出力ベクトルでないと判明したベクトルが除外されるにしたがい、徐々に大きな値をとる。一方、「歪みの上限値」 d_{upper} は探索の進行にしたがい徐々に小さな値へ更新される。この結果 d_{upper} と d_{lower} が一致すると探索は終了する。すなわち本手法によれば、コードブックベクトルの一部と歪み計算するだけで、最適出力ベクトルを見出すことができる。詳細な処理手順を図2に示す。

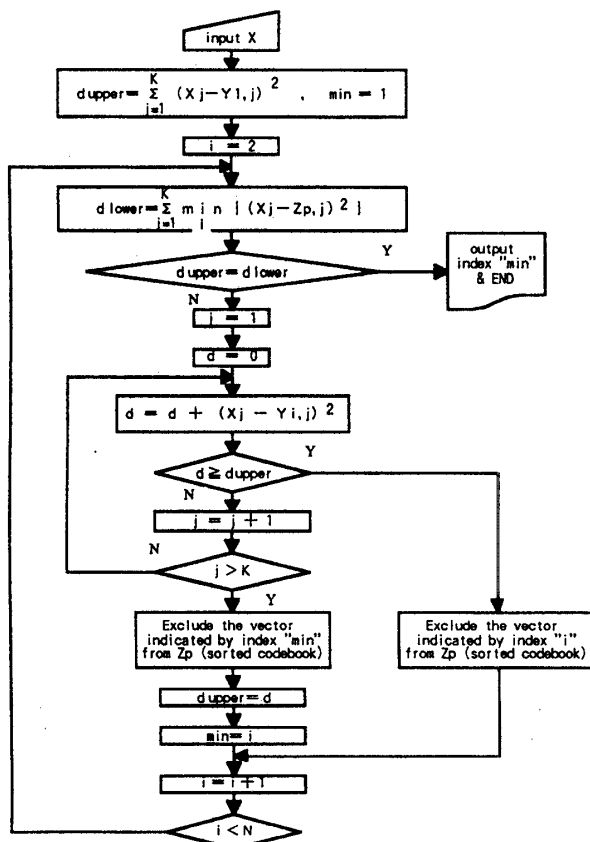


図2：提案手法のアルゴリズム

4. 結果と考察

提案した手法の有効性を検証するために、画像のベクトル量子化に必要な歪み計算回数と処理時間を計測した。用いた画像 (TV学会標準画像) は 256×256 画素で一画素当り8ビットとした。そしてこの画像を 4×4 のブロックに分割し、4096個のベクトルとした。このすべてのベクトルをトレーニング・シーケンスに用い、PNN法[4]により、 $N=1024, 512, 256, 128, 64, 32$ の各コードブックを作成した。各コードブックを用いて同じ画像をベクトル量子化し、その計算量を(0) 全探索法、(1) 「歪みの部分和」を用い、さらに歪み0となった時点で探索終了とするように改良した手

法 (「歪みの部分和」に基づく手法)、(2)提案手法とで比較した。

まず表1に歪み計算の対象となるベクトル数を示す。

表1：歪み計算の対象となるベクトル数

N	1024	512	256	128	64	32
V2	697499	275692	69968	55829	12386	4098
V2/V0[%]	16.63	13.15	6.67	10.65	4.72	3.13

$V0$ は全探索法で歪み計算の対象となるベクトルの数であり、この場合、 $V0 = 4096 \times N$ 、で求められる。

例えば $N=32$ のコードブックの場合、全探索法・「歪みの部分和」に基づく手法ともに、131072 (4096×32) 個のベクトルが歪み計算の対象となるのに対し、提案手法では4098個のベクトルに対して歪み計算をするだけで探索が終了する。すなわち提案手法では歪み計算の対象をコードブックベクトルの一部 (この場合3.13%) に限定することができる。

次に表2で歪み計算と処理時間についても比較する。

表2：歪み計算回数と処理時間

N	D1	D1/D0 [%]	T1[s]	D2	D2/D0 [%]	T2[s]
1024	18592130	27.70	29.803	3141058	4.68	27.642
512	13600263	40.53	20.054	1506507	4.49	13.897
256	10332509	61.59	14.599	550835	3.28	4.819
128	4951749	59.03	7.092	433669	5.17	4.449
64	3228458	76.97	4.626	169946	4.05	1.090
32	1506838	71.85	2.289	65568	3.13	0.871

$D0$ は全探索法による歪み計算回数であり、この場合、 $D0 = 16 \times 4096 \times N$ 、で求められる。処理時間 $T1, T2$ はSPARC Station 2で計測。

例えば $N=64$ のコードブックの場合、歪み計算回数は、「歪みの部分和」に基づく手法では全探索法の76.97%の3228458回であるが、提案手法では4.05%の169946回と非常に少ないことがわかる。これにより提案手法で高速なベクトル量子化が実現できることが示された。

5. おわりに

今後さらに高速化を図るために、本手法の並列処理について検討していく予定である。

参考文献

- [1] R. M. Gray, "Vector quantization," IEEE, ASSP Mag., pp. 4-29, Apr. 1984.
- [2] A. Buzo, A.H. Gray Jr., R.M. Gray, and J.D. Markel, "Speech coding based upon vector quantization," IEEE Trans. ASSP, Vol.28, No.10, pp.562-574, Oct. 1980
- [3] C.-D. Bei and R. M. Gray, "An Improvement of the Minimum Distortion Encoding Algorithm for Vector Quantization," IEEE Trans. Commun., COM-33, pp. 1132-1133, Oct. 1985
- [4] W.H. Equitz, "A New Vector Quantization Clustering Algorithm", IEEE Trans. ASSP, Vol.37, No.10, pp.1568-1575, Oct. 1989