AUGMENTING VECTOR QUANTIZATION WITH INTERVAL ARITHMETICS FOR IMAGE-CODING APPLICATIONS

Sandro Ridella, Stefano Rovetta, and Rodolfo Zunino

DIBE - Dept. Biophysical and Electronic Engineering - University of Genoa
Via all'Opera Pia 11a - 16145 Genova – Italy

ABSTRACT

Interval Arithmetics (IA) augments the basic Vector-Quantization (VQ) paradigm for image compression. The reformulated VQ schema allows prototypes to assume ranges of admissible locations rather than be clamped to specific space positions. The image-reconstruction process exploits the resulting degrees of freedom to make up for the excessive discretization (such as blockiness) that often affects VQ-based coding. The paper describes the algorithms for both the training and the run-time use of IAVQ codebooks; the possibility of data-driven training endows the proposed methodology with the flexibility and adaptiveness of standard VQ methods, as confirmed by experimental results on real images.

1. INTRODUCTION

The ability to attain considerable compression ratios in high-dimensional domains makes Vector Quantization (VQ) [1] suitable for image compression [2,3]. As to the quality of reconstructed images, quantizing the space into a few partitions often leads to an excessive discretization of represented data (blockiness). Removing such artifacts is still an open problem.

Interval Arithmetics (IA) [4] can be profitably integrated within the VQ paradigm; the major advantage is that reconstruction quality is enhanced without affecting compression performance. Interval-Arithmetics Vector Quantization (IAVQ) redefines VQ prototypes and lets them be placed in ranges of admissible locations rather than specific space positions. Thus a VQ prototype becomes an "interval prototype". A minimum-distance criterion rules codeword selection, hence the number of bits and the time for encoding a pixel block are the same as those of classical VQ. Interval prototypes at the receiver end provide the pixel-reconstruction process with degrees of freedom that improve image rendering. The final setting of pixel values results from an optimization process imposing a local-smoothness assumption; a Cellular Neural Network (CNN) [5,6] supports the final image-rendering process. Figure 1 presents a simplified schema of the methodology.

The IAVQ method formulates the image-reconstruction process as a constrained quadratic-programming problem [5] where interval codewords impose the bounds to the solution space. Such bounds are domain-adaptive, as IAVQ codebooks can be trained empirically by means of simple, fast algorithms. Thus IAVQ ensures the flexibility and example-driven ability of VQ schemata. As compared with classical low-pass filtering [6], interval quantities control the filtering action and ultimately prevent generalized blurring effects. The augmented model does not affect either compression ratio or speed performance. The circuitry supporting data coding uses standard-VQ hardware implementations [2], and the application of a CNN to the decoding process makes it possible to exploit well-known results in the HW literature [7,8].

The paper describes both the training process and the run-time use of IAVQ codebooks. Experimental results prove the notable performance of the augmented methodology.

2. THEORETICAL FRAMEWORK

The *basic VQ* schema associates with each *d*-dimensional point,

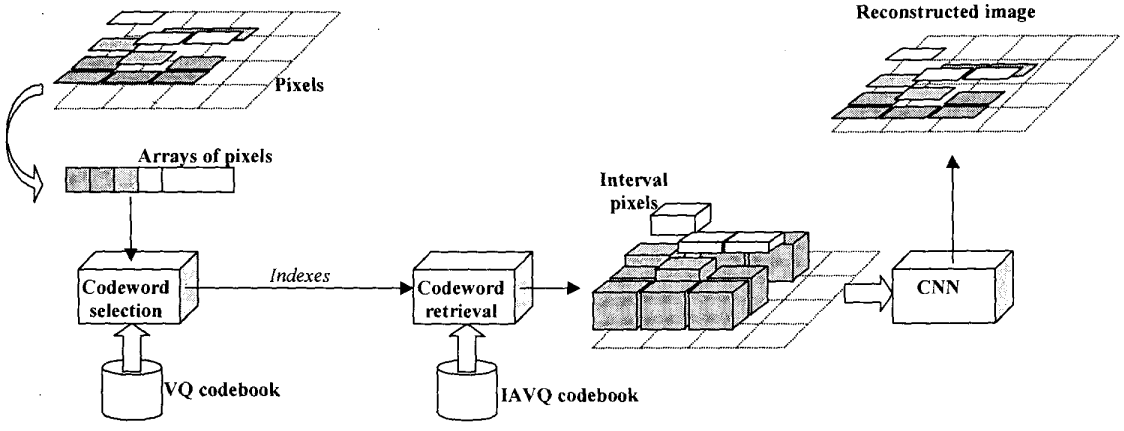

Figure 1. The IAVQ-based image-coding schema

0-7803-5482-6/99/$10.00 ©2000 IEEE

$\mathbf{x} \in \Re^d$, the best-matching element, $\mathbf{w}^*(\mathbf{x})$, selected from a codebook, $\psi = \{\mathbf{w}_j \in \Re^d,\ j = 1,\ldots,n_h\}$, such that:

$$\mathbf{w}^*(\mathbf{x}) = \arg \min_{\mathbf{w}_j \in \psi} \left\{ \sum_{i=1}^{d} \left(x^{(i)} - w_j^{(i)} \right)^p \right\} \qquad (1)$$

Several algorithms have been proposed to build up a suitable codebook for a given VQ-coding task. The research presented in this paper adopted a dedicated algorithm to assess both the number, n_h, and the positions of codevectors. This choice was mainly motivated by the availability of an efficient hardware implementation of the method [2]. VQ-based image compression in the pixel domain divides a picture into several (usually square) blocks representing coded samples, $\mathbf{x}_l \in \Re^d$, $l=1,\ldots,n_p$, where d is the number of pixels within a block, and n_p is the number of blocks making up the picture. Each block is encoded by working out its associate best-matching codevector according to (1). In the sample case of a standard, 8-bpp image that is split into blocks covering 8x8 pixels (d=64), a codebook holding n_h=256 codewords yields a compression ratio c_r=64.

Interval Arithmetics was introduced [4] as a handy formalism to treat quantities in the presence of uncertainty that makes exact determinations impossible. An interval variable defines a range of possible values for that quantity; any value within the interval is equally likely. An interval, X, is defined as an ordered pair $X = [x_L, x_U]$, such that $x_L, x_U \in \Re$ and $x_L \le x_U$. The bounds are admissible values for the associate quantity. In the following, by convention, uppercase letters will always denote interval quantities, whereas lowercase letters will stand for scalar variables. The interval formalism supports a compact algebraic notation in the space of intervals ($Y, A, B \in \Im, x \in \Re$), e.g.:

sum: $Y = A + B \Rightarrow Y = [a_L + b_L,\ a_U + b_U]$;

difference: $Y = A - B \Rightarrow Y = [a_L - b_U,\ a_U - b_L]$;

square: $Y = X^2 \Rightarrow Y = [\min\{x_L^2, x_U^2\}, \max\{x_L^2, x_U^2\}]$.

In comparison with classical algebra, some operations require additional computations to ensure the consistency of bounds. A d-dimensional interval vector is represented by an array of intervals: $\mathbf{W} = (W^{(1)}, \ldots, W^{(d)})$.

3. IAVQ-BASED IMAGE CODING

3.1 IAVQ codebook training

Codebook training requires determining the positions and ranges of all interval codewords. An exhaustive approach seems unrealistic because of the huge number of parameters. A simplified training strategy proceeds in two steps: first a standard VQ algorithm places prototypes in the domain space; then interval codewords are "inflated" around the VQ-generated initial positions. As the role of a codeword is to drive subsequent filtering by the CNN, the amplitude of each interval controls the extent of the associate degree of freedom. Intuitively, larger intervals should result in a better interpolation. In fact, zero-width intervals reduce the whole schema to basic, unfiltered VQ coding, but wider intervals, covering the whole range of possible pixel values, lead to unconstrained low-pass filtering. A data-driven training process can work out a tradeoff between such extremes empirically, thus making pixel reconstruction domain-adaptive.

The interval-width optimization starts from a trained VQ codebook $\psi = \{\mathbf{w}_j\}$. The only assumption is that ψ is adjusted by using an LBG-like algorithm [9]. LBG is an iterative codeword-positioning algorithm and guarantees that, at convergence, each prototype will be placed in the centroid of the associate partition. As pixels do not have preferential gray levels, the isotropic nature of the problem searches for a VQ-centered interval codeword, given by $\mathbf{W}_j = \left(W_j^{(1)}, \ldots, W_j^{(d)}\right)$ $W_j^{(i)} = [w_j^{(i)} - z_j^{(i)}, w_j^{(i)} + z_j^{(i)}]$. Training aims to determine the widest admissible ranges, $z_j^{(i)}$ (i=1,...,d, j=1,...,n_h).

A theoretical approach to the training problem leads to a set of Theorems giving analytical formulations of the interval ranges. Such amplitudes, however, result from a minimization process, hence nontrivial data distributions (e.g., pixel blocks) will yield very narrow ranges: a pair of samples lying close to a common boundary can squeeze the variation margin even if the remaining samples within each partition are tightly clustered. A more robust strategy is required in order to attain wider intervals.

The empirical approach retains the statistical distribution of data by considering each space dimension separately. The training method starts from a standard-VQ codebook trained with any LBG-like algorithm. Let $p\left(x^{(j,i)}\right)$ be the probability density function that describes the values assumed in the i-th dimension by the samples belonging to the j-th partition: $x^{(j,i)} \in \left\{x_l^{(i)} \text{such that } \mathbf{x}_l \in \pi_j\right\}$. The picture pixel depth bound these values: $x_{MIN} \le x^{(j,i)} \le x_{MAX}\ \forall i \forall j$. The robust algorithm computes the histogram of the observed values for each space dimension. The interval amplitude is set by imposing that the resulting range include a given share of the covered samples. The IAVQ codebook comprises, for each codeword, an interval and its "reference" value (the original VQ codeword position).

IAVQ training algorithm

1. Input: $\psi = \{\mathbf{w}_j\ j=1,\ldots,n_h\}$: VQ LBG codebook;
   $\pi_j, j=1,\ldots,n_h$; sample partitions spanned by ψ;
   $0 < t \le 1$; the requested coverage of data values.
2. For each partition $\pi_j, j=1,\ldots,n_h$
   1.a For each dimension $i = 1,\ldots,d$
      1.a.I Evaluate the histogram, $h\left(x^{(j,i)}\right)$, of $x^{(j,i)}$
      1.a.II Define: $p\left(x^{(j,i)}\right) = h\left(x^{(j,i)}\right) / n_p^{(j)}$
      1.a.III Work out the amplitude, $z>0$, such that:

$$\int_{x_L^{(j,i)}}^{x_U^{(j,i)}} p\left(x^{(j,i)}\right) dx = t\ ;\ x_U^{(j,i)} = \min\left\{x_{MAX}, w_j^{(i)} + z\right\};$$

$$x_L^{(j,i)} = \max\left\{x_{MIN}, w_j^{(i)} - z\right\}.$$

   1.b Assemble the codeword $\mathbf{W}_j = \left(W_j^{(1)}, \ldots, W_j^{(d)}\right)$ as
   $W_j^{(i)} = [x_L^{(j,i)}, x_U^{(j,i)}]$, $i = 1,\ldots,d$
2. Output: the interval codebook, $\Psi = \left\{\left(\mathbf{W}_j, \mathbf{w}_j\right) j = 1,\ldots,n_h\right\}$.

"Peak" densities, reflecting concentrated distributions of values, will determine narrow intervals, whereas shallow densities will

yield wider intervals. Space partitions with high concentrations of samples will be represented by "narrow" intervals, ultimately witnessing a higher level of confidence in the prototype position. Space regions covering scattered samples will be represented by "wider" and more uncertain interval codewords.

3.2 IAVQ-based image coding

The basic idea of the image-compression method is to leave the compression strategy unaffected (i.e., supported by a standard VQ codebook and WTA competition). The encoder considers only the "reference" codeword positions, and scalar distances are involved in WTA selection. The additional interval information is used in the decoding process at the receiver end. The image-coding algorithm just repeats its VQ counterpart:

IAVQ-based image-coding algorithm

0. Input: Image blocks, $\beta=\{\mathbf{x}_1,\ldots,\mathbf{x}_B\}$
   VQ codebook, ψ ; Index set, $\theta = \varnothing$
1. For each block $\mathbf{x} \in \beta$
   2.a Work out $\mathbf{w}^*(\mathbf{x}) = \arg\min_{\mathbf{w}\in\varphi}\left\{\|\mathbf{w}-\mathbf{x}\|^2\right\}$;
   let q^* be the index of $\mathbf{w}^*(\mathbf{x})$ in ψ;
   2.b Set $\theta = \theta \cup \{ q^* \}$
3. Output: the set of codeword indexes, $\theta = \{q_1,\ldots, q_B\}$

The description of the compressed picture is the same for VQ and IAVQ. Such an approach virtually decouples the image coder and the image decoder. IAVQ becomes a superset of VQ, hence the receiver acts as an augmented version of a classical VQ decoder. Moreover, the overall compression performance remains unaffected, as the compression ratio keeps constant. Finally, the computational cost brought about by plugging in IAVQ is entirely supported by the receiver.

3.3 IAVQ-based image decoding

The image decoder retrieves interval codewords from received indexes. IAVQ-based image reconstruction requires selecting the values that maximize picture quality from the intervals associated with image pixels. Mean Square Error is the analytical measure of distortion, and pixel estimation is formalized as a constrained MSE-minimization problem. The first constraint takes into account the contiguity of pixels, and imposes that the values of adjacent pixels be not sensibly different. The smoothness assumption brings low-pass filtering into the reconstruction action. The second constraint requires the values of reconstructed pixel not to exceed the associate interval bounds. By this mechanism, IAVQ introduces a balancing control that inhibits uniform low-pass filtering.

A Cellular Neural Network [5,6] supports the reconstruction process: build up a CNN with the same planar structure as the reconstructed image. Let $y(r,c)$ denote the reconstructed pixel value on the image coordinates (r,c), and let $y_{VQ}(r,c)$ be the associate reference value, given by the coordinates of the original VQ prototype. Each network cell is characterized by an internal status variable, $u(r,c)$ [5]; the nonlinear function yielding the output activation $y(r,c)$ (i.e., the corresponding pixel value) is given by [6]:

$$y(r,c)=\frac{1}{2}\left[\left|u(r,c)-y_L(r,c)\right|-\left|u(r,c)-y_U(r,c)\right|+\left(y_L(r,c)+y_U(r,c)\right)\right] \quad r=1,\ldots,m,\ c=1,\ldots,n \qquad (2)$$

where m and n are the number of rows and columns in the image, respectively, and $y_L(r,c)$ and $y_U(r,c)$ are the lower and upper bounds, respectively, of the interval representing the pixel at the position (r,c). As compared with the standard CNN model [6], the lower and upper saturation levels in the cell non-linearity are determined, for each cell, by the lower and upper bounds of the interval IAVQ prototype encoding the specific location. This sets a limit on the generalized low-pass filtering effect by preventing unconstrained fluctuations of pixel values.

The MSE-based problem formulation implies a quadratic optimization problem, the solution of which must be found within the (hyper)box bounded by interval ranges. This leads to a class of very complex problems, and in principle one might question the use of CNN to this purpose. The theory presented in [5], however, gives an effective and efficient algorithm to tackle quadratic-programming problems. A theoretical analysis shows that the image-reconstruction formalism and the cell model (2) can be proved to be equivalent to the problem setting used in [5]. The reconstruction algorithm can be outlined as follows:

IAVQ-based image-decoding algorithm

0. Input: Set of codeword indexes, $\theta = \{q_1,\ldots, q_B\}$
   Interval codebook, Ψ
1. Build a planar CNN with m rows and n columns
2. Set $a = \sqrt{d}$; $b_R = m / a$; $b_C = n / a$;
3. For each index $k = 1,\ldots,B$
   3.a Retrieve the indexed interval codeword $\mathbf{W}_{q_k} \in \Psi$
   3.b Set $b_x = \lfloor k / b_c \rfloor$; $b_y = (k-1) \bmod b_R$;
   3.c For each dimension $i = 1,\ldots,d$
      3.c.I Set $r = a \cdot b_x + \lfloor i / a \rfloor$; $c = a \cdot b_y + (i-1) \bmod a$;
      3.c.II Set $y_{VQ}(r,c) = \overline{w}_{q_k}^{(i)}$; $y_L(r,c) = w_{q_k,L}^{(i)}$;
      $y_U(r,c) = w_{q_k,U}^{(i)}$;
4. Run the CNN ruled by (2) according to the algorithm [5].
5. Output: the final set of pixel values $\{ y(r,c) \}$.

Most of the receiver computational cost depends on the CNN evolution process, which yields the reconstructed pixel values. Thanks to both the standard features of the neural structure and the general validity of the optimization algorithm, the method fully benefits from the huge literature about effective HW implementations of CNNs [7,8].

4. EXPERIMENTAL RESULTS

The experimental validation of the presented methodology involved a comparison between standard CNN-filtered images and those obtained by the IAVQ reconstruction method. Both numerical (MSE plots) and qualitative results are reported, showing the superior performance deriving from the interval-based control of low-pass filtering. In all the experiments, unconstrained filtering resulted in a far larger MSE at the convergence time; instead, the distortion curves associated with the IAVQ method always exhibited a saturation trend, settling at constant values. The latter phenomenon witnesses the limiting effect of IAVQ bounds on cell nonlinearities, as they prevented uncontrolled pixel variations and inhibited image degradation.

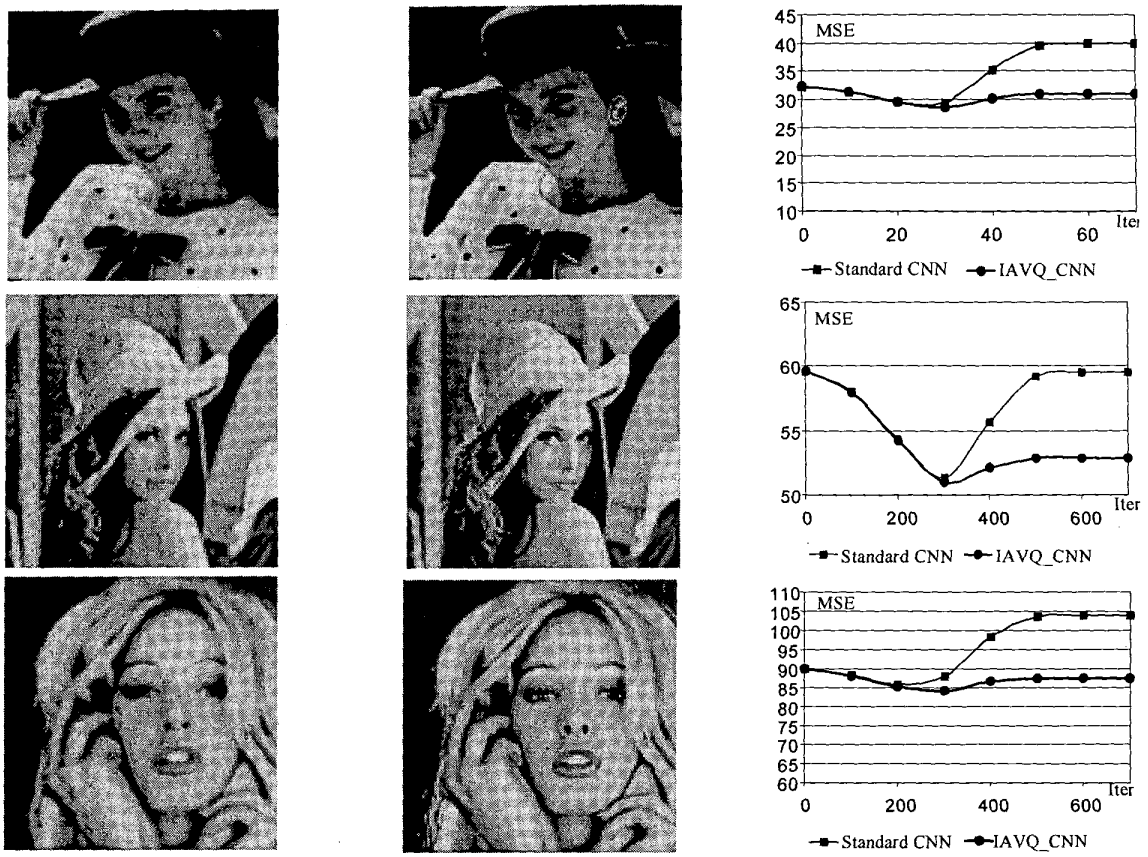

Figure 2. Comparison of standard CNN and IAVQ-based image reconstruction
left: results from standard CNN low-pass filtering; center: results from IAVQ; right: comparative plots of associate MSEs

Unconstrained filtering yields images suffering from apparent blurring effects. By contrast, images decoded by the IAVQ method take advantage of low-pass filtering, which contributes to removing blockiness, but local details are preserved.

5. REFERENCES

[1] *IEEE Trans Inf.Theory*, Spec.Issue on Vector Quantization, March 1982, vol.IT-28, No.2.

[2] Rovetta S, Zunino R "Efficient training of Neural Gas vector quantizers with analog circuit implementation" *IEEE Trans. on Circ. Sys II*, June 1999, vol.46, No.6, pp.688-698.

[3] Nasrabadi N, King R "Image Coding Using Vector Quantization:A Review", *IEEE Trans. Commun.*, Aug. 1988, pp. 957-971

[4] Alefeld G, Herzberger J *Introduction to interval computation*, 1983, Academic Press, NY

[5] Gilli M, Civalleri PP, Roska T, Chua LO " Analysis of time-varying cellular neural networks for quadratic global optimization" *Int. J. Circ. Theory and Apps.*, 1998, vol.26, pp.109-126

[6] Shi BE, Chua LO "Resistive grid image filtering: Input/Output analysis via the CNN framework" *IEEE Trans. Circ. Systems I*, July 1992, vol.39, No.7, pp.531-548

[7] Kinger P, Steyaert M "A programmable analog Cellular Neural Network CMOS chip for high speed image processing" *IEEE J. Solid State Circ.*, Mar 1995, vol. 30, No.3, pp.235-243.

[8] Salerno M, Sargeni F, Bonaiuto V "A dedicated multi-chip programmable system for Cellular Neural Networks" *Analog Int. Circ. and Sig.Proc.*, 1999, vol.18, pp.277-288

[9] Linde Y, Buzo A, Gray RM "An Algorithm for vector quantizer design", *IEEE Trans. Commun.*, Vol. COM-28, pp. 84-95, Jan. 1980.