

# IAVQ—Interval-Arithmetic Vector Quantization for Image Compression

Sandro Ridella, *Member, IEEE*, Stefano Rovetta, *Member, IEEE*, and Rodolfo Zunino, *Member, IEEE*

**Abstract**—Interval arithmetic (IA) can enhance vector quantization (VQ) in image-compression applications. In the interval arithmetic vector quantization (IAVQ) reformulation of classical VQ, prototypes assume ranges of admissible locations instead of being clamped to specific space positions. This provides the VQ-reconstruction process with some degrees of freedom, which do not affect the overall compression ratio, but help make up for coarse discretization effects. In image compression, IA attenuates artifacts (such as blockiness) brought about by the VQ schema. This paper describes the algorithms for both the training and the run-time use of IAVQ. Data-driven training endows the methodology with the adaptiveness of standard VQ methods, as confirmed by experimental results on real images.

**Index Terms**—Author, please supply index terms. E-mail keywords@ieee.org for more info.

## I. INTRODUCTION

VECTOR QUANTIZATION (VQ) [1] encodes information by means of a set of prototypes (codewords) in the observed domain. Each point in the data space is represented by the codeword that maximizes a similarity criterion. Compression stems from using a codebook whose (log) cardinality is smaller than the number of bits describing a datum. The fact that entire space partitions are encoded by the associated codewords causes VQ to be an efficient compression method. The heavy computational load represents a crucial issue of VQ schemata. It can be tackled by either dedicated hardware circuitry [2]–[6] or accelerated algorithms [7]–[9].

The index-based coding schema allows VQ to attain considerable compression ratios in high-dimensional domains. This makes the method suitable for image compression [10]–[13]. Thanks to the possibility of locally representing scene contents, VQ coding schemata may perform effectively in specific applications that involve image understanding [14] and multimedia data processing [15]. When very low bit-rate compression must be attained, VQ techniques, as well as other nonconventional approaches (e.g., wavelets), may represent a valid alternative to standard compression algorithms such as Joint Photographers Expert Group (JPEG) [14]–[16] in specific applications.

When considering the quality of reconstructed images, it is well-known that quantizing the space into a few partitions may

Manuscript received September 1999; revised August 2000. This work was supported in part by the Italian Ministry for University and Scientific and Technological Research (MURST). This paper was recommended by Associate Editor A. Skodras.

The authors are with the Department of Biophysical and Electronic Engineering (DIBE), University of Genoa, 16145 Genova, Italy (e-mail: ridella@dibe.unige.it; rovetta@dibe.unige.it; zunino@dibe.unige.it).

Publisher Item Identifier S 1057-7130(00)11029-8.

lead to an excessive discretization of represented data. This may give rise to undesired effects, such as blockiness. Removing artifacts is still an open problem [17]–[21]. Any additional information transmitted to remove image defects may tend to increase the required bandwidth, hence a myriad of deblurring approaches have been proposed to improve image quality at the decoder end without increasing the number of bits sent.

This paper shows that interval arithmetic (IA) [22] can be profitably integrated within the VQ-based paradigm. The major advantage is that reconstruction quality is enhanced without affecting compression performance. Interval arithmetic 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.” Nevertheless, the number of bits and the time for encoding a datum are the same as those of classical VQ coding.

Interval prototypes at the receiver end provide the pixel-reconstruction process with degrees of freedom, thus permitting a qualitative improvement in image rendering. Evaluating the method quantitatively is complicated by the lack of a valid model of visual perception, hence the current research adopts mean square error as a standard distortion measure. An approach using the quantization interval has been proposed for JPEG-compressed images [18], [23], where projections onto convex sets (POCS) constrain the reconstruction to be consistent with the encoded bitstream subject to a smoothness constraint. From a regularization perspective, a frequency-domain method to remove blocking from JPEG-encoded pictures led to a gradient-based quadratic-programming problem [17].

Likewise, IAVQ formulates the image-reconstruction process as a constrained quadratic-programming problem [24], where interval codewords impose bounds to the solution space. The final optimization process implies a local-smoothness assumption. In place of the gradient-based method used in [17], a cellular neural network (CNN) [25], [26] drives the regularization task and supports the actual image rendering. The major advantage of adopting a CNN lies in the notable efficiency of the eventual circuit implementation. The reconstruction method at the decoder end differs significantly from classical low-pass filtering [19], [27], as interval quantities strictly control (as much as slopes in [17]) the filtering action and ultimately prevent generalized blurring effects. In IAVQ, the bounds are domain-adaptive, as the used codebooks can be trained empirically by simple and fast algorithms. Thus IAVQ ensures the example-driven ability of VQ schemata and keeps their flexibility.

The augmented model does not affect either the compression ratio or the speed performance. The circuitry supporting data-coding uses standard VQ hardware implementations [6].

The application of a CNN to the decoding process exploits well known results reported in the literature, showing that CNN structures can solve quadratic-optimization problems effectively. In addition, the planar, locally interconnected architectures of CNNs strongly simplify hardware circuitry [28]–[31].

This paper describes both the training process and the run-time use of IAVQ codebooks. The image-coding reconstruction performance of the method is evaluated in comparison with well known rendering approaches that can be applied after standard VQ coding, namely, classical low-pass filtering and unconstrained CNN-based low-pass filtering. Experimental results show the effectiveness of the proposed methodology and provide numerical evaluations and qualitative visual assessments.

The paper is organized as follows: Section II briefly summarizes VQ information coding and gives a basic formalism for IA, then the actual IAVQ method is presented. Section III deals with the interval codebook-training problem. Section IV describes the use of IAVQ for image coding, and especially details its operation at the receiver end by means of a CNN structure. Section V reports the experimental results obtained and compares the method performances with those of related approaches. Finally, some concluding remarks are made in Section VI.

## II. FRAMEWORK FOR IAVQ

### A. VQ-Based Data and Image Coding

The basic VQ schema adopted here associates with each  $d$ -dimensional point,  $\mathbf{x} \in \mathcal{R}^d$ , the best-matching element,  $\mathbf{w}^*(\mathbf{x})$ , selected from a codebook,  $\psi = \{\mathbf{w}_j \in \mathcal{R}^d, j = 1, \dots, n_h\}$ , such that:

$$\mathbf{w}^*(\mathbf{x}) = \arg \min_{\mathbf{w}_j \in \psi} \left\{ \sum_{i=1}^d (x^{(i)} - w_j^{(i)})^2 \right\}. \quad (1)$$

Several algorithms have been proposed to build up a suitable codebook for a given VQ-coding task, involving a vast literature on pattern recognition [1], [32] and neural networks [33], [34]. The research presented in this paper adopted a plastic version [35] of the “neural gas” model [33] to assess both the number,  $n_h$ , and the positions of codevectors. This choice was mainly suggested by the availability of an efficient hardware implementation of the method [6].

VQ-based image compression has been used in the literature in both the transform domain and the spatial domain. In the transform domain, VQ applies to frequency-based representations of coded images (typically, discrete cosine transform) [13]. In the spatial domain, it operates directly on image pixels, and will be adopted as a default in the following. This choice results in a lower computational cost, as images need not be transformed, and it simplifies real-time applications [14].

Pixel-domain VQ divides a picture into (usually square) blocks representing coded samples,  $\mathbf{x}_l \in \mathcal{R}^d, l = 1, \dots, n_p$ , 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 associated best-matching codevector (1). For

images with gray-level pixels that may assume up to  $2^b$  values, the resulting compression ratio is given by

$$c_r = \frac{b \cdot d}{\lceil \log_2 n_h \rceil}. \quad (2)$$

For example, an 8-bpp image that is split into blocks covering  $8 \times 8$  pixels ( $d = 64$ ), a (typical) codebook holding  $n_h = 256$  codewords yields a compression ratio  $c_r = 64$ . Clearly, larger block sizes yield higher compression ratios. On the other hand, the codebook size might represent a severe problem if too large blocks are encoded, due to the involved memory requirements and the increased computational load. As a consequence, relatively small block sizes are used (typically  $4 \times 4$  or, in some nonstandard cases,  $8 \times 8$ ).

In practice, several technical mechanisms have been added to the basic schema to improve image appearance. Mean residual coding (MRC) [36] subtracts from each block its mean value before VQ encoding. This mechanism separates brightness information in advance and lets VQ represent contrast. Adaptive block splitting (ABS) [37], [38] allows the encoder to choose various block sizes in compliance with local image contents. Picture areas containing little information are encoded by larger blocks, whereas smaller blocks cover detail-rich image regions. Both MRC and ABS can notably improve the quality of a reconstructed image, at the cost of transmitting additional information (block mean values and quadtree structure, respectively). These add-ons by no means affect the basic coding schema (1), which will keep a general validity in the following.

### B. Interval Arithmetic Formalism

Interval Arithmetic was introduced [22] as a handy formalism to treat quantities in the presence of uncertainty or noise that makes exact determinations impossible. An interval variable defines a range of possible values for that quantity; any value within the interval is admissible and equally likely. An interval  $X$  is defined as an ordered pair  $X = [x_L, x_U]$ , such that  $x_L, x_U \in \mathcal{R}$  and  $x_L \leq x_U$ . The bounds are admissible values for the associated 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,  $\mathcal{I}$ . The following operations are defined ( $Y, A, B \in \mathcal{I}, x \in \mathcal{R}$ ):

sum

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

difference

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

translation

$$Y = A + x \Rightarrow Y = [a_L + x, a_U + x];$$

amplification

$$Y = x \cdot A \Rightarrow Y = [x a_L, x a_U] \quad \text{if } x \geq 0;$$

$$Y = [x a_U, x a_L] \quad \text{if } x < 0;$$

product

$$Y = A * B \Rightarrow Y = [\min\{a_L b_U, a_U b_U, a_L b_L, a_U b_L\},$$

$$\max\{a_L b_U, a_U b_U, a_L b_L, a_U b_L\}];$$

square

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

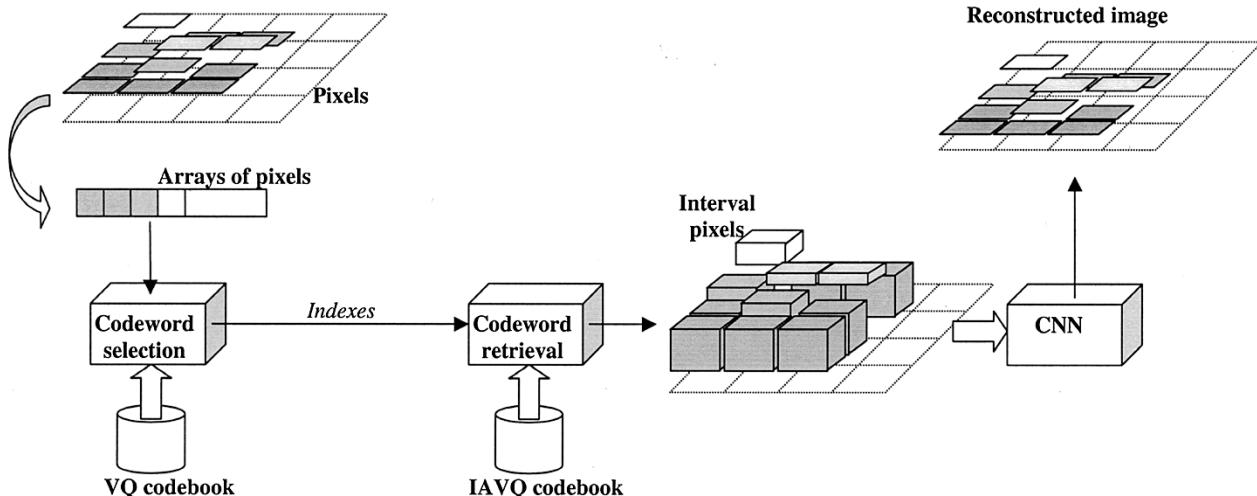


Fig. 1. The IAVQ-based image-coding schema.

In comparison with classical algebra, some operations require additional computations to ensure the consistency of bounds. The case of the square function exemplifies an IA peculiarity for nonmonotonic functions, as the interval function  $X^2$  may differ from  $X * X$ , if  $X$  includes the origin.

A  $d$ -dimensional interval vector is represented by an array of intervals:  $\mathbf{W} = (W^{(1)}, \dots, W^{(d)})$ . The Euclidean distances from  $\mathbf{W}$  to a vector  $\mathbf{x} \in \mathbb{R}^d$  and another interval vector  $\mathbf{X} \in \mathcal{I}^d$  are given by

$$\begin{aligned} D(\mathbf{W}, \mathbf{x}) &= \sum_{i=1}^d (W^{(i)} - x^{(i)})^2 \\ D(\mathbf{W}, \mathbf{X}) &= \sum_{i=1}^d (W^{(i)} - X^{(i)})^2. \end{aligned} \quad (3)$$

### C. Concept and Basics of IAVQ

The basic idea underlying IAVQ is to extend the flexibility of VQ schemata. A crucial drawback of VQ-based representation of information is that the encoding process results in a single location of the data space (a codeword), which represents every point in the associated partition. Thus, compression is often obtained at the cost of coarseness. The purpose of augmenting VQ by IA is to extend the meaning of “codeword” by encoding a partition of the data space with a range of admissible space locations. In practice, a set of intervals replaces the single centroid coordinates. This redefines the concept of a “prototype,” which can be regarded as a “hyper-box” spanning an interval of valid coordinates along each dimension. Such a mechanism clearly introduces degrees of freedom into the reconstruction process, and the specific application requirements will suggest the suitable criterion to select the appropriate coordinates within the box.

In VQ-based picture compression, codeword coordinates have a one-to-one mapping to image pixels, hence the outcome of IAVQ block coding is a block of “interval pixels.” IAVQ-based image coding exploits the variability associated with each interval to compensate for the coarseness brought

about by the quantization process. The crucial advantage of the VQ-derived schema is that both the position and the extension of each hyperbox can be made domain-adaptive by learning from examples. Thus, the overall compression and rendering schema preserves the flexibility of basic VQ.

The selection of the eventual pixel values stems from a regularization process that minimizes a quadratic cost function within the constraints set by interval codewords. The inherent smoothness constraint aims to exploit the contiguity of neighboring pixels, and leads to a filtering action that involves a quadratic-programming optimization problem. This crucial task can be effectively accomplished by a cellular neural network, whose major features are its theoretically established convergence properties and a planar architecture allowing efficient hardware implementations. Fig. 1 shows a schematic representation of the process.

### III. IAVQ TRAINING

In principle, the codebook-training problem requires that one determine the positions and ranges of all interval codewords at the same time. Such an exhaustive approach, however, seems computationally unfeasible because of the huge number of parameters to be estimated. The simpler training strategy adopted in this research proceeds in two steps: 1) a standard VQ algorithm places prototypes in the domain space and 2) interval codewords are “inflated” around the VQ-generated initial positions. Such an approach has the following two basic reasons: 1) splitting the optimization process into a twofold process reduces training complexity and 2) the VQ coding schema still applies and is comprehended as a special case of the more general IAVQ.

The amplitude of each interval controls the extent of the associated degree of freedom, hence one might intuitively expect that larger intervals result in a better interpolation. Indeed, using zero-width intervals reduces the whole schema to basic VQ coding, but the widest intervals covering the whole range of possible values  $[0, 2^b - 1]$  lead to unconstrained filtering, which often proves unsatisfactory. The required tradeoff between such extremes can be worked out empirically by data-driven training.

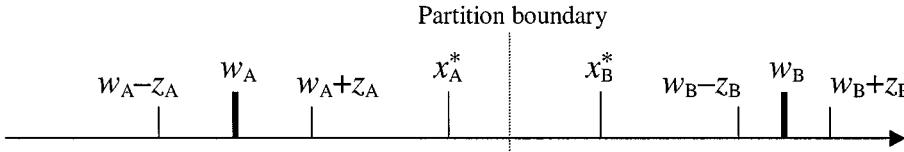


Fig. 2. Geometrical properties of sample distribution determine interval amplitudes (one-dimensional case).

### A. Theoretical Approach to IAVQ Codebook Training

The theoretical approach to IAVQ codebook training derives analytical values for the ranges of each interval prototype, under the basic constraint that the interval settings should not generate ambiguities as to the categorization of any training sample. Thus, the data partitions set by VQ regions will remain unaffected. Let  $\pi_j$  denote the data partition associated with the  $j$ th codeword, i.e., the set of samples belonging to the partition covered by  $\mathbf{w}_j$ :  $\pi_j = \{\mathbf{x}_l, l = 1, \dots, n_p^{(j)} \text{ such that } \mathbf{w}^*(\mathbf{x}_l) = \mathbf{w}_j\}$ .

In compliance with the above training strategy, interval-width optimization starts from a trained VQ codebook  $\psi = \{\mathbf{w}_j\}$ . The only assumption made in this phase is that  $\psi$  is adjusted by using an LBG-like algorithm [32]. The LBG model, often also known as  $k$ -means, is an iterative codeword-positioning algorithm, and is adopted here because it guarantees that, at convergence, each prototype will eventually be placed in the centroid of the associated partition.

Pixels are not assumed to have preferential gray levels, hence the isotropic nature of the problem makes it possible to search for a VQ-centered interval codeword that may be expressed as  $\mathbf{W}_j = (W_j^{(i)}, \dots, W_j^{(d)})$ ;  $W_j^{(i)} = [w_j^{(i)} - z_j^{(i)}, w_j^{(i)} + z_j^{(i)}]$ . The training process aims to determine the widest admissible ranges,  $z_j^{(i)}$  ( $i = 1, \dots, d, j = 1, \dots, n_h$ ). The analysis will initially consider one-dimensional situations, then the result will be extended to the general case of multidimensional codewords.

**Lemma 1:** Let  $w_A$  and  $w_B$  be VQ scalar prototypes positioned by an LBG-like algorithm [32]. The optimal half-amplitudes,  $z_A$  and  $z_B$ , of the associated intervals that preserve VQ-based partitions are

$$z_A = z_B = |x_B^* - x_A^*|/2$$

where

$$x_A^* = \arg \min_{x \in \pi_A} \{|x - w_B|\}$$

and

$$x_B^* = \arg \min_{x \in \pi_B} \{|x - w_A|\}.$$

**Proof:** Assume, without loss of generality, that  $w_A < w_B$ . The LBG-like algorithm ensures that a VQ prototype lies in the barycentre of its data partition; therefore, there exist  $x_A \in \pi_A, x_B \in \pi_B$  such that  $w_A \leq x_A < x_B \leq w_B$ . By definition,  $x_A^*(x_B^*)$  is the sample belonging to the partition  $\pi_A(\pi_B)$  that lies closest to  $w_B(w_A)$ . For this property,  $w_A \leq x_A^* < x_B^* \leq w_B$ . The situation is sketched in Fig. 2. Prototype  $w_A$  can take on any value in the range  $[w_A - z_A, w_A + z_A]$ . In order to ensure a correct categorization of sample  $x_A^*$  in the worst case, the following condition must be fulfilled:

$$|w_A - z_A - x_A^*| < |w_B - z_B - x_A^*| \quad (4)$$

which can be rewritten as

$$w_A + w_B > 2x_A^* + (z_A + z_B). \quad (5)$$

By analogy, the dual condition for the second prototype leads to

$$w_A + w_B < 2x_B^* - (z_A + z_B). \quad (6)$$

By subtracting (5) from (6), one obtains

$$z_A + z_B < x_B^* - x_A^*. \quad (7)$$

If one now assumes that the available amplitude is uniformly split between the two intervals, imposing  $z_A = z_B$  in (7) proves the assertion. **Q.E.D.**

The previous Lemma points out that the amplitudes of an interval prototype depend on the relative positions of the samples lying closest to the partition boundaries. This property can be extended to the multidimensional case by considering, for each codeword, the set of neighboring prototypes in the Voronoi tessellation of the space and by applying Lemma 1 to the connecting lines. It follows that any standard VQ codebook can be “inflated” to a nonnull interval codebook. The empirical distribution of data will drive the extent of the process. Such a property can be formally expressed as follows:

**Theorem 1:** Let  $\psi = \{\mathbf{w}_j \mid j = 1, \dots, n_h\}$  be a VQ codebook obtained by an LBG-like algorithm and spanning an unambiguous data partition. Then there exists an interval codebook,  $\Psi = \{\mathbf{W}_j \mid j = 1, \dots, n_h\}$ , spanning the same partitions set by  $\psi$  and having nonnull interval amplitudes.

**Proof:** Consider the  $j$ th element of  $\psi$ . From computational geometry, work out the set of prototypes that share a partition boundary with  $\mathbf{w}_j$ . Denote by  $\nu_j = \{\mathbf{w}_{jn} \mid n = 1, \dots, n_h^{(i)} \leq n_h\}$  such a set of neighbors of  $\mathbf{w}_j$ , and by  $\mathbf{u}_{jn}$  the versor connecting  $\mathbf{w}_j$  to  $\mathbf{w}_{jn}$

$$\mathbf{u}_{jn} = \frac{\mathbf{w}_{jn} - \mathbf{w}_j}{\|\mathbf{w}_{jn} - \mathbf{w}_j\|}, \quad n = 1, \dots, n_h^{(i)}. \quad (8)$$

By projecting onto  $\mathbf{u}_{jn}$  the two data partitions spanned by the two prototypes,  $\pi_j$  and  $\pi_{jn}$ , respectively, one obtains the situation illustrated in Fig. 3, in which the pair of “critical” samples is worked out as

$$\begin{aligned} x_{jn}^* &= \max_{\mathbf{x}_l \in \pi_j} \{\mathbf{u}_{jn} \bullet (\mathbf{x}_l - \mathbf{w}_j)\} \\ x_{nj}^* &= \max_{\mathbf{x}_l \in \pi_{jn}} \{\mathbf{u}_{jn} \bullet (\mathbf{w}_{jn} - \mathbf{x}_l)\} \end{aligned} \quad (9)$$

where  $\bullet$  indicates the scalar dot product. By applying Lemma 1 to the resulting monodimensional case, one obtains the widest range of variation,  $z_{jn}$ , for  $\mathbf{w}_j$  in the direction  $\mathbf{u}_{jn}$

$$z_{jn} = \frac{\|\mathbf{w}_{jn} - \mathbf{w}_j\| - (x_{nj}^* + x_{jn}^*)}{2}. \quad (10)$$

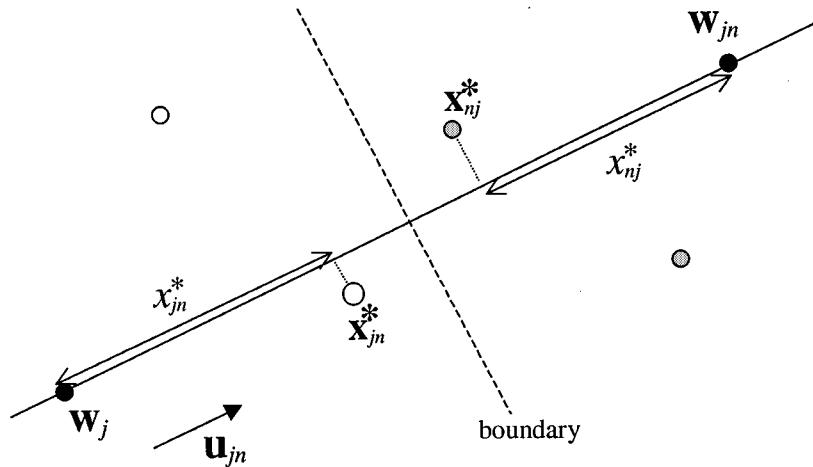


Fig. 3. Peculiarities of sample distribution constrain interval amplitudes (multidimensional case).

By hypothesis, the partitioning schema set by the VQ codebook is not ambiguous, hence there are no samples lying at the boundary between two partitions. As a consequence, the margin attained by (10) is nonnull

$$z_{jn} > 0 \quad \forall j \forall n. \quad (11)$$

Expression (10) makes it possible to work out a variation vector,  $\mathbf{z}_{jn}$ , defined as

$$\mathbf{z}_{jn} = z_{jn} \mathbf{u}_{jn} = (z_{jn} u_{jn1}, \dots, z_{jn} u_{jnd}). \quad (12)$$

The set of vectors obtained by (12) give the maximum variabilities of prototype  $\mathbf{w}_j$  in the directions of all neighbors. Each vector imposes a constraint on the allowable positions of  $\mathbf{w}_j$  such that the original partition is preserved in that specific direction. A choice for the interval amplitudes of prototype  $\mathbf{W}_j$  is given by

$$\mathbf{z}_j = \left( \min_{\mathbf{w}_{jn} \in \pi_j} \{z_{jn} r_{jn1}\}, \dots, \min_{\mathbf{w}_{jn} \in \pi_j} \{z_{jn} r_{jnd}\} \right). \quad (13)$$

The interval settings (13) satisfy all constraints (12), hence they preserve the partition  $\pi_j$  relevant to the  $j$ -th prototype. In order to prove that the obtained interval codewords are nonnull, from (8) we derive

$$\sum_{i=1}^d u_{jni}^2 = 1 \quad \forall j \forall n. \quad (14)$$

By combining (11) with (14), one can verify that there exist nonnull interval amplitudes, thus completing the proof. Q.E.D.

### B. Operational Approach to IAVQ Codebook Training

Theorem 1 motivates interval codewords from a theoretical point of view, but it might set severe limitations on the practical usefulness of the resulting codebook. In particular, the preservation of VQ partitions might turn out to be too strict a constraint on the codebook-construction process. The interval amplitudes (13) result from a minimization process, hence one expects that nontrivial distributions of real data (e.g., pixel blocks) will give rise to very narrow margins (10) and very small amplitudes (13). Such a phenomenon often occurs due to singularities

in the spatial distribution of data: a pair of samples lying close to a common boundary can squeeze the variation margin (10), even if the remaining samples within each partition are tightly clustered. Imposing partition consistency on interval setting leads to a very sensitive process. A more robust strategy is required in order to attain wider intervals, featuring some resistance against outliers or peculiar sample configurations.

The empirical approach proposed in this paper aims to retain the statistical distribution of data. The basic idea is to consider each space dimension separately. Like the previous algorithm, the training method starts from a standard-VQ codebook trained by any LBG-like algorithm. The partitions of data points associated with each codeword provide the statistical sample for the analysis.

The training algorithm can be outlined as follows: let  $p(x^{(j,i)})$  be the probability density function that describes the values assumed in the  $i$ th dimension by the samples belonging to the  $j$ th partition:  $x^{(j,i)} \in \{x_l^{(i)} \text{ such that } \mathbf{x}_l \in \pi_j\}$ . In image-coding applications, these values are bounded by the current pixel depth  $x_{\text{MIN}} \leq x^{(j,i)} \leq x_{\text{MAX}} \forall i \forall j$ . In the case of 8-bpp gray-level images,  $x_{\text{MIN}} = 0$  and  $x_{\text{MAX}} = 255$ . In order to support robust statistics, the algorithm computes the histogram of the observed values for each space dimension. The interval amplitude is eventually determined by requiring that the resulting range include a given share of the covered samples. Thus, the final IAVQ codebook comprises, for each codeword, an interval and its “reference” central value given by the original VQ codeword position.

#### a) IAVQ training algorithm:

- 1) Input:  $\psi = \{\mathbf{w}_j | j = 1, \dots, n_h\}$ ; VQ codebook trained by an LBG-like algorithm;  $\pi_j, j = 1, \dots, n_h$ ; the partitions of samples spanned by  $\psi$ ;  $0 < t \leq 1$ ; the requested coverage of data values.
- 2) For each partition  $\pi_j, j = 1, \dots, n_h$ 
  - 1.a For each dimension  $i = 1, \dots, d$ 
    - 1.a.I Evaluate the histogram,  $h(x^{(j,i)})$ , of values  $x^{(j,i)} \in \{x_l^{(i)} \text{ such that } \mathbf{x}_l \in \pi_j\}$

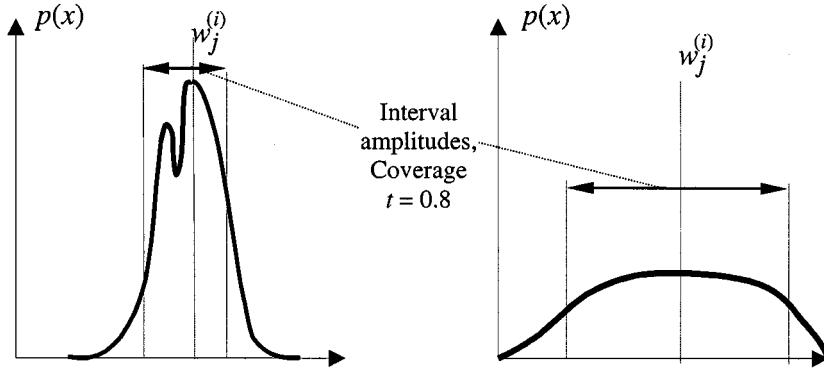


Fig. 4. Wider distributions of pixel values yield larger intervals in codeword dimensions.

1.a.II Estimate the probability density function:

$$p(x^{(j,i)}) = h(x^{(j,i)})/n_p^{(i)}$$

1.a.III Work out the amplitude,  $z > 0$ , such that:

$$\int_{x_L^{(j,i)}}^{x_U^{(j,i)}} p(x^{(j,i)}) dx = t$$

$$x_U^{(j,i)} = \min\{x_{\text{MAX}}, w_j^{(i)} + z\}$$

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

1.b Assemble the interval codeword

$$\mathbf{W}_j = (W_j^{(1)}, \dots, W_j^{(d)})$$

$$W_j^{(i)} = [x_L^{(j,i)}, x_U^{(j,i)}], \quad i = 1, \dots, d.$$

3) Output: the interval codebook,

$$\Psi = \{(\mathbf{W}_j, \mathbf{w}_j), j = 1, \dots, n_h\}.$$

The described algorithm is quite simple and involves a lower computational load, as compared with the theoretical approach (9) and (10). The crucial step 1.a.III can be easily performed by progressively inflating the average-centered interval until the expected coverage is attained.

Peak densities, reflecting concentrated distributions of values, will determine narrow intervals, whereas shallow densities, suggesting widespread distributions of values, will yield wider intervals. The situation is exemplified in Fig. 4. As a result, space partitions with high concentrations of samples will be represented by “narrow” interval codewords, ultimately witnessing a higher level of confidence in the prototype position. Conversely, space regions covering scattered samples will be represented by “wider” interval codewords, suggesting a larger uncertainty in the optimal location of the prototype. The possibility of absorbing information about the spatial distribution of samples within the associated IAVQ prototype actually represent an additional interesting feature of the overall training method.

#### IV. IAVQ RUN-TIME OPERATION

##### A. IAVQ-Based Image Coding

The codeword-building method just described does not guarantee that the partitioning schema spanned by the original VQ codebook will be preserved. The histogram-based method generates intervals at the single-dimension level and

independently of one another, hence the variability regions associated with prototypes may (actually, they are quite likely to) overlap. As a result, in order to match a (vector) sample against an (interval) codeword, the winner has to be selected by comparing interval distances. Choosing the best matching candidate by observing overlapping interval distances might prove not trivial. This strongly suggests that reformulating the VQ-coding problem *verbatim* with the interval formalism may lead to possibly unnecessary complications.

The basic idea underlying IAVQ image coding is to leave the compression strategy unaffected (i.e., supported by a standard VQ codebook and WTA competition). The encoder uses only “reference” codeword positions (yielding scalar distances), whereas interval information is taken into account only at the decoder end. Thus, the IAVQ image-coding algorithm just repeats its VQ scalar counterpart.

##### b) IAVQ-based image coding:

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

The above approach offers several benefits. First, using standard VQ encoding does not require additional training in the image-coding subsystem, and the resulting bitstream describing the compressed picture remains unchanged. This guarantees a full compatibility with any noninterval VQ-based compression method, and preserves the space partitions spanned by the original VQ codebook. Such an approach decouples the image-coder and the image-decoder processes, and the receiver acts as an augmented version of a classical VQ decoder. Secondly, the overall system’s compression ratio remains constant, as the encoder need not know the features embedded in the decoding module. Finally, the additional computational cost brought about by plugging in IAVQ is entirely supported by the receiver. Therefore, the available circuitry can be exploited for VQ encoding, which relieves the IAVQ method of the need for specific hardware design.

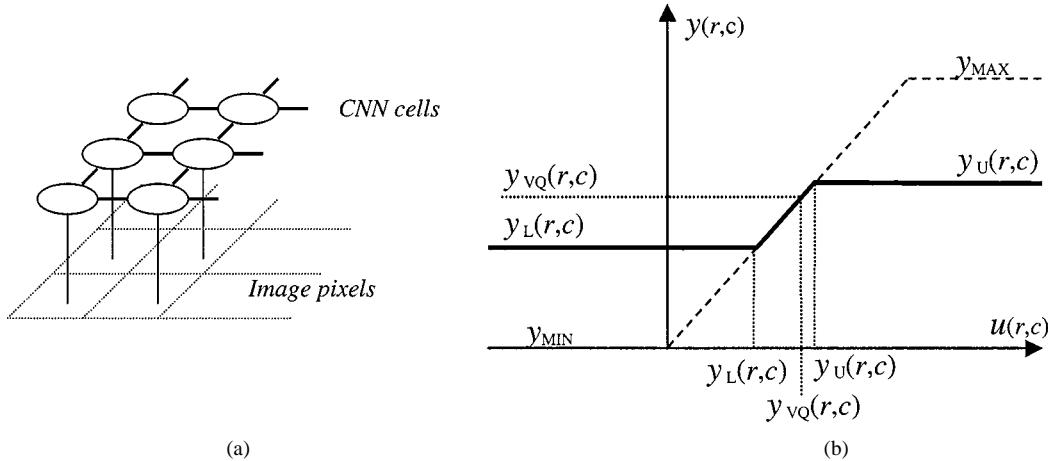


Fig. 5. Using CNN-based filtering for IAVQ decoding. (a) The planar structure of the network. (b) Codeword intervals restrain each cell's nonlinearity.

### B. IAVQ-Based Image Decoding

Image decoding consists in retrieving and reassembling the set of codewords associated with the blocks of a transmitted image. As explained in Section II-C, pixels are represented by interval quantities (Fig. 1), and IAVQ-based reconstruction requires that their final values be selected from the associated ranges to optimize picture quality. The centroid positions characterizing the VQ codebook are derived by an unsupervised training algorithm that minimizes MSE. As explained in Section III-B, they are the centroid, “reference” positions of the IAVQ codewords, which are inflated to create validity ranges. All values within an interval are equally likely *a priori* but, in the case of image reconstruction, it seems reasonable that the eventual IAVQ result should privilege consistency with the original VQ encoding. Therefore, the basic term in the cost function penalizes the displacement from the reference scalar position, whereas the information from surrounding pixels provides external conditioning. Thus, pixel estimation can be formalized as a quadratic minimization problem subject to some constraints.

The first constraint takes into account the contiguity of pixels, and requires that the values of adjacent pixels be not sensibly different. This (typical) assumption about image smoothness limits the distribution of spatial frequencies in the image. In principle, one might use well-known deblurring methods that implement low-pass filtering after VQ reconstruction. Such techniques, however, would not exploit the variability information provided by interval quantities. The difference between a pixel value and the average value of its neighbors originates a regularization term in the cost function.

The second constraint derives from the interval-based codewords, and requires the final pixel values not to exceed the associate interval bounds. By this mechanism, IAVQ introduces a balancing action that counteracts uniform low-pass filtering. In the approach presented in this paper, such a constraint is not expressed explicitly in the cost function. Actually, it is buried in the saturated nonlinearity characterizing each cell of a CNN. The use of CNN structures [25], [26] for IAVQ has some basic advantages. First, the cost function is expressed uniformly throughout the image, that is, its global value results from the sum of individual contributions from all pixels, whereas specific bounds at the pixel level are considered locally. A simpler ex-

pression for the cost also facilitates optimization. Secondly, the regularization task involves a quadratic-programming problem that can be tackled by a CNN effectively. Finally, CNNs offer a planar, distributed, loosely interconnected architecture that greatly simplifies hardware implementations.

IAVQ might be regarded as a kind of deblurring method, although the method actually uses interval bounds to prevent blurring. The major advantage of IAVQ lies in driving the regularization-based reconstruction of image pixels, whereas a CNN plays a basic role in accomplishing the optimization task effectively. So one could summarize that IAVQ yields the actual qualitative improvement, whereas CNNs provide the method with effective and fast convergent performance.

### C. CNN-Based Image Rendering

The constrained-minimization problem calls for a dedicated structure supporting the related heavy computational load. The pixel-reconstruction process is reformulated following the problem-setting presented in [27]. Let us build up a CNN with the same planar structure as the reconstructed image. The cells show a one-to-one correspondence with the image pixels [see Fig. 5(a)]. Let  $y(r,c)$  denote the reconstructed pixel value on the image coordinates  $(r,c)$ , and let  $y_{ref}(r,c)$  be the associated reference value. In the filtering structure [27],  $y_{ref}(r,c)$  was given by the original image pixel. In the present context, such a value is obviously unknown to the decoder, hence the “central” value of the associated interval is used instead. This value is associated with each codeword by the training algorithm described in Section III, and is given by the coordinates of the original VQ prototype. For the sake of simplicity, such a quantity is denoted by  $y_{VQ}(r,c)$ . With these notations, the cost function to be minimized [27] is expressed as

$$\begin{aligned} E = \lambda & \sum_{r=0}^{m-1} \sum_{c=0}^{n-1} (y(r,c) - y_{VQ}(r,c))^2 \\ & + \sum_{r=0}^{m-1} \sum_{c=0}^{n-2} (y(r,c) - y(r,c+1))^2 \\ & + \sum_{r=0}^{m-2} \sum_{c=0}^{n-1} (y(r,c) - y(r+1,c))^2 \end{aligned} \quad (15)$$

where  $m$  and  $n$  are the numbers of rows and columns in the image, respectively, and  $\lambda$  is a positive factor. The first term biases a reconstructed pixel toward its expected value; the other summations bring in a regularizing action that involves adjacent pixels, thus incorporating the smoothness constraint into the reconstruction process. The quantity  $\lambda$  controls the intensity of the network smoothing action:  $\lambda = 0$  imposes a maximum smoothing, and  $\lambda \gg 1$  minimizes the contributions of neighboring pixels and inhibits the consequent low-pass filtering effect.

Each network cell is characterized by an internal status variable,  $u(r, c)$  [24]. The nonlinear function yielding the output activation  $y(r, c)$  (i.e., the corresponding pixel value) is given by [27]

$$\begin{aligned} y(r, c) = & \frac{1}{2} [|u(r, c) - y_L(r, c)| - |u(r, c) - y_U(r, c)| \\ & + (y_L(r, c) + y_U(r, c))], \\ r = 1, \dots, m, c = 1, \dots, n \end{aligned} \quad (16)$$

where  $y_L(r, c)$  and  $y_U(r, c)$  are the lower and upper bounds, respectively, to the interval representing the pixel at the position  $(r, c)$ . Expression (16) exhibits a slight, yet important, difference, as compared with a standard CNN model, like that adopted in [27]. In the cell nonlinearity, the lower and upper saturation levels are not constant and equal for all the cells (in the case of 8-bbp images,  $y_{\text{MIN}} = 0$  and  $y_{\text{MAX}} = 255$ ), but they are determined, for each cell, by the lower and upper bounds to the interval provided by the IAVQ prototype encoding the specific location [see Fig. 5(b)]. Thus, the range information about each codeword dimension (and the associated pixel) affects the behavior of the corresponding cell. The adaptiveness buried in the distributed structure actually implements the second constraint on the optimization process, as it sets a limit on the generalized low-pass filtering effect by preventing unconstrained fluctuations of pixel values.

From an analytical perspective, the cost formulation (15) under the constraint (16) implies a quadratic optimization problem, the solution of which must be found within the (hyper)box bounded by interval ranges. Such a complex problem would in general be very difficult to solve analytically, and in principle, one might question the choice of a CNN for that purpose. In order to determine the progression of the internal status value of each cell, from (15) we derive the following quantity:

$$\begin{aligned} \frac{\partial E}{\partial y(r, c)} = & (\lambda + 4)y(r, c) - \lambda y_{\text{VQ}}(r, c) - y(r-1, c) \\ & - y(r+1, c) - y(r, c-1) - y(r, c+1) \end{aligned} \quad (17)$$

which is the analytical counterpart of the resistive-grid circuit equation presented in [27]. If we now write the cell status-update rule as

$$\frac{du(r, c)}{dt} = -u(r, c) + \left[ -\frac{\partial E}{\partial y(r, c)} + y(r, c) \right] \quad (18)$$

expression (18) turns out to be equivalent to the quadratic-form minimization formulated in [24, p. 110, eq. (2)].

As a consequence, the image-reconstruction cellular network fully complies with the formalism described in [24]. Thus, one can benefit from the results obtained by that research, which provided an effective CNN-based algorithm that the authors



Fig. 6. Artificial test image.

proved to solve the specific class of quadratic programming problems, like the one involved in IAVQ-based image rendering.

The optimization method described in that paper is theoretically guaranteed to converge to the minimum of the cost function. The method was therefore adopted in the present research to control the CNN progress. The overall image-reconstruction algorithm can be outlined as follows:

c) *IAVQ-based image reconstruction:*

- 0) Input: Set of codeword indexes,  $\theta = \{q_1, \dots, q_B\}$   
representing image blocks  
Interval codebook,  $\Psi$
- 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  $k = 1, \dots, B$ 
  - 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  $i = 1, \dots, d$ 
    - 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  $y_{\text{VQ}}(r, c) = \bar{w}_{q_k}^{(i)}$   
 $y_L(r, c) = w_{q_k, L}^{(i)}$   $y_U(r, c) = w_{q_k, U}^{(i)}$
- 4) Run the CNN ruled by (16) and (18) according to the algorithm described in [24].
- 5) Output: the final set of pixel values  
 $\{y(r, c); r = 0, \dots, m-1; c = 0, \dots, n-1\}$ .

#### D. Circuit Implementation of IAVQ

The core of the computational cost of the IAVQ-based coding methodology lies in the codeword selection for the encoding process. The compatibility of the block-classification schema described in Section III-B with standard VQ makes it possible to exploit the results of previous research. In particular, the whole image-coding process can be supported by a dedicated VLSI device [2]–[6] that can perform the WTA-based prototype matching and selection.

Most of the computational cost at the receiver end is instead associated with the CNN evolution process, which yields the

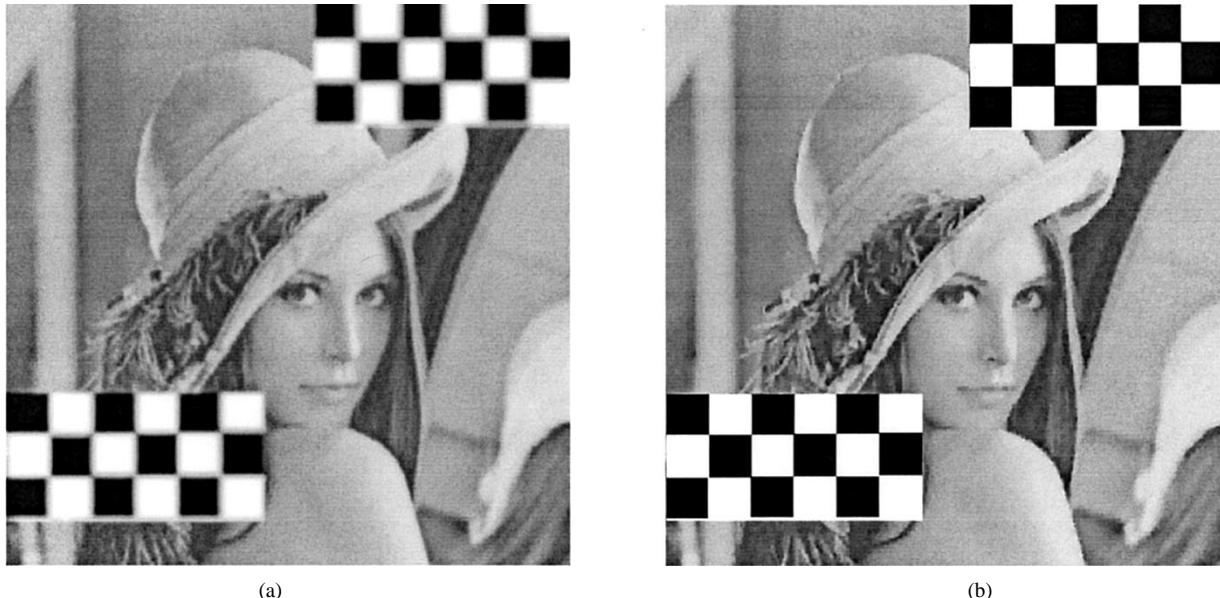


Fig. 7. IAVQ reconstruction can support selective low-pass filtering to preserve sharp edges. (a) Standard low-pass filtering. (b) IAVQ decoding.

reconstructed pixel values. Thanks to both the standard features of the neural structure and the general validity of the optimization algorithm implemented, the present research fully benefits from the vast literature on effective hardware implementations of CNNs [28]–[31]. An in-depth analysis of many possible alternatives is beyond the scope of this paper, which is mainly focused on the effective use of IA to enhance VQ-based image coding. The truly important advantage of using CNNs at the decoder end is that the distributed structure can be efficiently implemented in a planar hardware architecture, thus ensuring convergence speed for real-time performance in the image-rendering process.

## V. EXPERIMENTAL RESULTS

The empirical validation process involves two setups to evaluate the effectiveness of the IAVQ method. The first uses artificial pictures to highlight the advantages of the IAVQ representation. The second testbed involves pictures generally used in image-compression experiments, and compares the IAVQ reconstruction performance with that resulting from standard low-pass and unconstrained CNN filtering. The comparison is made both quantitatively by evaluating the MSEs in the various cases and qualitatively by visually assessing the reconstruction results.

### A. Artificial Testbed Results

The artificial experiment was suitably designed to highlight the effects of interval-controlled low-pass filtering. The training set included both real, gray-level images and artificial patterns with large black and white regions. The combination of the two types of pictures resulted in a peculiar IAVQ codebook. The real images with uniform distributions of gray levels yielded codewords with wide intervals, whereas the peak distributions characterizing the synthetic patterns gave codewords with very narrow intervals. The resulting IAVQ codebook covered both situations depicted in Fig. 4. The results obtained on a critical

test picture containing sharp edges (Fig. 6) confirmed the substantial difference between unconstrained and IAVQ-controlled image filterings. The application of standard, unbounded CNNs to the VQ-decoded picture conveyed an undifferentiated blurring, which affected visual quality significantly, especially in reconstructing sharp edges. Similar results were obtained by applying classical low-pass filtering.

By contrast, the IAVQ-based method preserved critical boundaries. Checkerboard elements were encoded by the “narrow” interval codewords, whose limited ranges prevented the embedded low-pass filter from bringing in unconstrained averaging. The reconstructed edges maintained their original sharpness, whereas the remaining, “natural” image regions were encoded by the wider codewords allowing a more accurate reconstruction. Fig. 7 presents reconstruction results, and enables one to make a visual comparison of the two methods.

### B. Real-Image Coding Experiments

The experiments on normal, gray-level pictures aimed at assessing the actual reconstruction effectiveness of the method, and allowed both a quantitative and a qualitative evaluation. The training phase consisted in developing a VQ codebook from a set of ordinary pictures, and then in building an associate IAVQ codebook by using the method described in Section III.

The block size was set to  $4 \times 4$  pixels to make the method compatible with standard VQ systems. Moreover, in order to highlight the actual contribution of IAVQ and to avoid that image-enhancement methods might interfere with the measurements, no specific technique (e.g., mean residual coding or adaptive block splitting) was used in the tests; such an experimental choice, however, did not affect the general validity of the approach. Since pixel depth was eight bits in the used images, the overall compression ratio was  $C_r = 16 \times 8 / \log 2(N_h)$ , where  $N_h$  is the number of prototypes in the codebook.

In the test phase, a set of images not used for training were VQ-encoded. As expected, when decoded by standard VQ

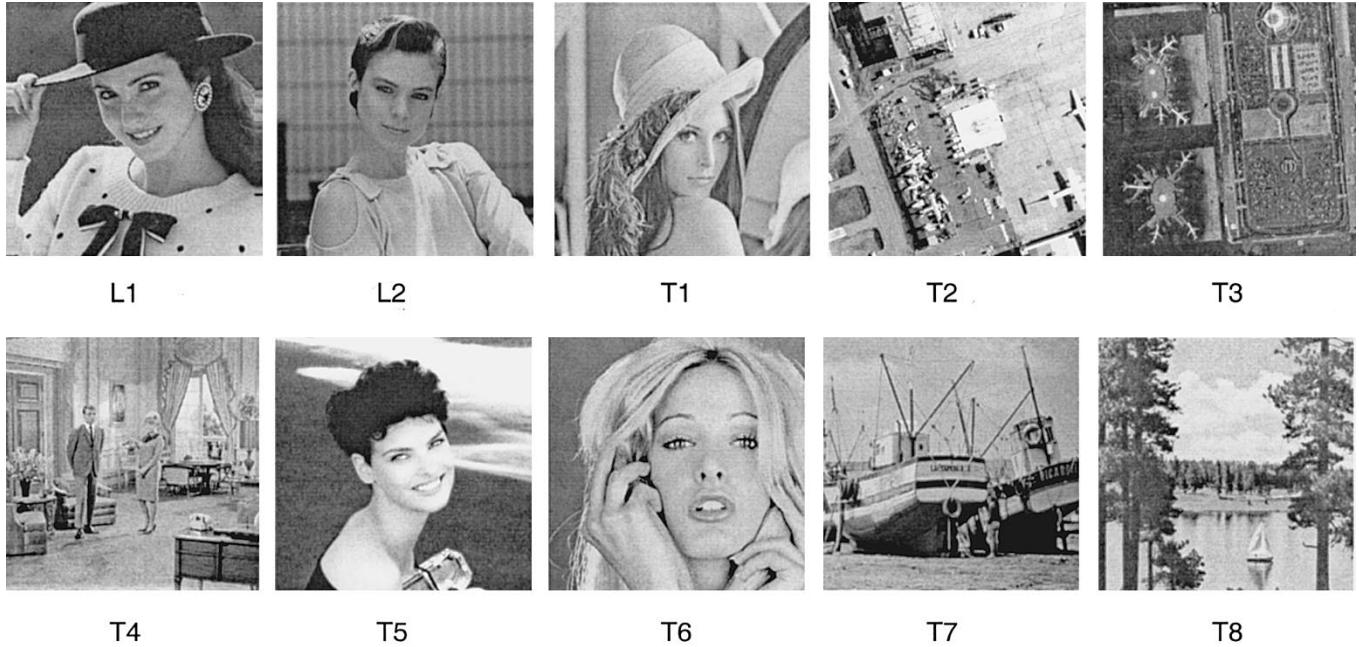


Fig. 8. Experimental image set. The two top-leftmost pictures constitutes the training set.

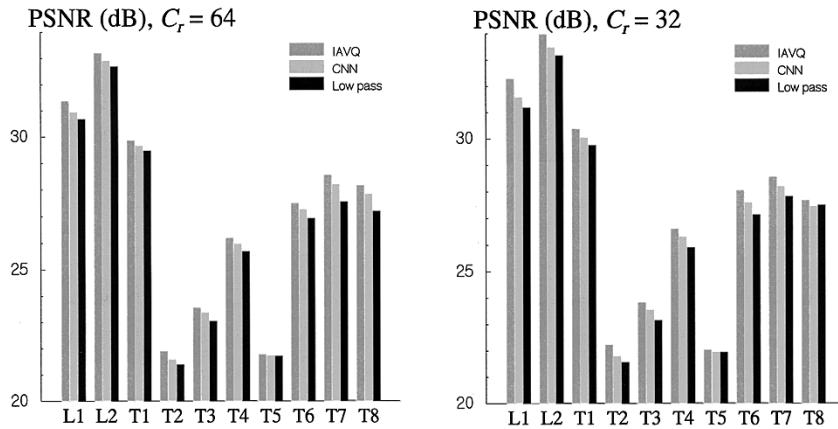


Fig. 9. Comparative evaluation of the reconstruction methods performances.

the resulting pictures suffered from apparent blockiness. In order to verify the advantages of interval representation, the VQ-decoded pictures were processed by different deblocking algorithms to allow a comparative evaluation. According to a classical deblocking approach [19], first the pictures underwent low-pass filtering. Alternatively, a standard CNN was chosen to perform unconstrained low-pass processing [27]. The results obtained were compared with those obtained by the IAVQ reconstruction algorithm described in Section IV. This experimental setup was repeated for different compression ratios, that is, for different values of  $N_h$ . As the compression ratios remained the same for each set of tests using: 1) standard low-pass filtering; 2) CNN-based low-pass filtering; or 3) IAVQ-based reconstruction, this made it possible to evaluate the specific contribution of IA-based rendering in a comparative way.

The pictures used in the experiments are presented in Fig. 8, showing both training ( $L_x$ ) and test ( $T_x$ ) images. The experimental setup seems quite valid, as some test samples differ sensibly from the training ones. The tests involved different set-

tings for the codebook cardinality, which was increased from  $N_h = 64$  ( $C_r = 64$ ) up to  $N_h = 256$  ( $C_r = 16$ ). Experimental evidence showed that the benefit of IAVQ rendering varies with the involved compression ratio. More precisely, if  $C_r$  is not very high, the VQ conventional approach does not bring in any blocking artifact, nor does it provide any blurring effect. Conversely, the effect of IAVQ-based regularization becomes more and more visible when the compression ratio increases. The experiments using different settings of  $C_r$  just aimed to establish the framework within which the IAVQ technique must be applied, and the conclusion was that IAVQ benefits can be appreciated for  $C_r > 16$  ( $N_h < 256$ ).

The numerical evaluation of reconstruction results is given in terms of measured PSNR. Fig. 9 presents a summary of such numerical measurements. In all the experiments, the IAVQ reconstruction method proved better than its counterparts. In fact, in some cases, numerical differences in the observed distortions may not appear very marked, yet the steadily better performance of IAVQ is an indirect confirmation of its efficacy.



Fig. 10. Samples of reconstructed test images. Left: IAVQ; right: low-pass filtered VQ.

A reliable validation of the method effectiveness, however, can be obtained by visually comparing reconstruction results.

Unconstrained filtering yielded images suffering from apparent blurring effects. By contrast, images decoded by the IAVQ

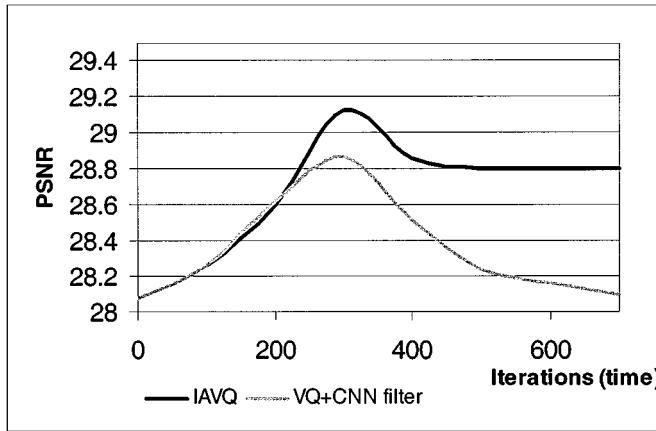


Fig. 11. Samples of run-time progression curves during CNN convergence.

method took advantage of low-pass filtering to remove blockiness, but local details and features were preserved, thus giving the overall pictures an evident better appearance. Unbounded CNN-based filtering exhibited an intermediate performance between the two extrema. As expected, the gap in visual appearance among the different methods shrinks when the compression ratio decreases, until no visible effects can be noticed ( $C_r = 16$ ). The pictures presented in Fig. 10 give samples of the different performances of the reconstruction methods.

A deeper insight into the method's operation can be gained by observing the PSNR progressions during the CNN run-time convergence for the IAVQ method and the unbounded regularization process. In all the experiments, like the one illustrated in Fig. 11, the distortion curves associated with the IAVQ method always exhibited a saturation trend, settling at constant values, whereas unbounded filtering exhibited no cutoff behavior. Such a phenomenon witnesses the limiting effect of IAVQ bounds on cell nonlinearities, as they prevented uncontrolled pixel variations and ultimately inhibited image degradation.

An important issue related to IAVQ effectiveness concerns the involved computational load. Clearly, the cost associated with any classical enhancement method based on filtering will compare favorably with the cost brought about by the CNN convergence process when simulated via software modeling. In fact, the approach presented in this paper cannot prescind from a target circuit implementation supporting the CNN operation. In such a case, the computational advantage of a standard filter reduces significantly due to the distributed nature of the network convergence. It is anyway worth stressing that regularization-based approaches to image enhancement will typically lead to a quadratic-programming problem. This is the case, for example, with the frequency-based method presented in [17], where the optimization task was tackled by a gradient-projection method that might prove cumbersome for hardware implementation. To sum up, from the previous results one might conclude that the higher computational cost associated with IAVQ seems to be the price for a better performance in image reconstruction. The ultimate choice will clearly depend on the requirements for the specific application considered.

## VI. CONCLUSION

The integration of the basic VQ paradigm for image compression with interval-based representation is a significant novel aspect of the research described in this paper. The resulting methodology exploits interval quantities to control the image-rendering process that enhances the eventual image quality. Therefore, the IAVQ model can benefit from both the possibility of low-pass compensation for undesired artifacts and the data-driven adaptiveness provided by empirical training of IAVQ codebooks. The reported results on a real generalization testbed also confirm the method's practical effectiveness.

However, the presented research is some way from being conclusive, as several crucial aspects still seem worth investigating. First of all, the proposed codebook-construction algorithm is simple enough for a straightforward implementation, but it might surely benefit from more sophisticated models of pixel-value distributions. The described image-compression process, instead, appears as the most promising approach for the overall framework, especially in view of its simplicity and compatibility with standard VQ image-coding systems. It also seems unlikely that the adopted CNN optimization algorithm will be liable to further significant improvements, mainly thanks to its sound theoretical framework and general validity.

The primary feature of the proposed IAVQ method, however, lies in its full compatibility with standard research in basic information-coding areas such as those of vector quantization and cellular neural networks. As a result, the image-coding methodology proposed in this paper can immediately exploit the progress made in those specific areas, with direct advantages in terms of efficient hardware implementations.

## REFERENCES

- [1] Special Issue on Vector Quantization, *IEEE Trans. Inform. Theory*, vol. IT-28, Mar. 1982.
- [2] G. Cauwenberghs and V. Pedroni, "A low-power CMOS analog vector quantizer," *IEEE J. Solid-State Circuits*, vol. 32, pp. 1278–1283, Aug. 1997.
- [3] W. C. Fang, C. Y. Chang, B. J. Sheu, O. T. C. Chen, and J. C. Curlander, "VLSI systolic binary tree-searched vector quantizer for image compression," *IEEE Trans. VLSI Syst.*, vol. 2, pp. 33–44, Feb. 1994.
- [4] K. Tsang and B. W. Y. Wei, "A VLSI architecture for a real-time code book generator and encoder of a vector quantizer," *IEEE Trans. VLSI Syst.*, vol. 2, pp. 360–364, June 1994.
- [5] W. C. Fang, B. J. Sheu, O. T. C. Chen, and J. Choi, "A VLSI neural processor for image data compression using self-organization networks," *IEEE Trans. Neural Networks*, vol. 3, pp. 506–518, June 1992.
- [6] S. Rovetta and R. Zunino, "Efficient training of neural gas vector quantizers with analog circuit implementation," *IEEE Trans. Circuits Syst. II*, vol. 46, pp. 688–698, June 1999.
- [7] E. Vidal, "An algorithm for finding nearest neighbors in (approximately) constant average time," *Pattern Recognit. Lett.*, vol. 4, pp. 145–157, July 1986.
- [8] C. H. Lee and L. H. Chen, "Fast closest codeword search algorithm for vector quantization," *Proc. Inst. Elect. Eng.*, vol. 141, no. 3, pp. 143–148, June 1994.
- [9] R. Jain, A. Madisetti, and R. L. Baker, "An integrated circuit design for pruned tree-search vector quantization encoding with an off-chip controller," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, pp. 147–158, Mar. 1992.
- [10] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Norwell, MA: Kluwer, 1992.
- [11] N. Nasrabadi and R. King, "Image coding using vector quantization: A review," *IEEE Trans. Commun.*, vol. 36, pp. 957–971, Aug. 1988.
- [12] R. Dony and S. Haykin, "Neural network approaches to image compression," *Proc. IEEE*, vol. 83, pp. 288–303, Feb. 1995.

- [13] C. Amerijck, M. Verleysen, P. Thissen, and J. D. Legat, "Image compression by self-organizing kohonen map," *IEEE Trans. Neural Networks*, vol. 9, pp. 503–507, June 1998.
- [14] S. Rovetta and R. Zunino, "Vector quantization for license-plate location and image coding," *IEEE Trans. Ind. Electron.*, vol. 35, Feb. 2000.
- [15] S.-Y. Kung and J.-N. Hwang, "Neural networks for intelligent multimedia processing," *Proc. IEEE*, vol. 86, pp. 1244–1272, June 1998.
- [16] C. Constantinescu and J. A. Storer, "Improved techniques for single-pass adaptive vector quantization," *Proc. IEEE*, vol. 82, pp. 933–939, June 1994.
- [17] S. Minami and A. Zakhori, "An optimization approach for removing blocking effects in transform coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, pp. 74–82, Apr. 1995.
- [18] A. Zakhori, "Iterative procedures for reduction of blocking effects in transform image coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, pp. 91–95, Mar. 1992.
- [19] H. Reeve and J. S. Lim, "Reduction of blocking effects in image coding," *Opt. Eng.*, vol. 23, no. 1, pp. 34–37, Jan. 1984.
- [20] M.-Y. Shen, J.-W. Kim, and C.-C. J. Kuo, "Fast compression artifact reduction technique based on nonlinear filtering," in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. IV, Orlando, FL, 1999, pp. 179–182.
- [21] V. Solo, "Selection of regularization parameters for total variation denoising," in *Proc. IEEE Int. Conf. Acou. Speech Sig. Proc. ICASSP'99*, vol. III, Phoenix, AZ, pp. 1653–1655.
- [22] G. Alefeld and J. Herzberger, *Introduction to Interval Computation*. New York: Academic, 1983.
- [23] S. Hein and A. Zakhori, "Halftone to continuous-tone conversion of error-diffusion coded images," *IEEE Trans. Image Proc.*, vol. 4, pp. 208–215, Feb. 1995.
- [24] M. Gilli, P. P. Civalleri, T. Roska, and L. O. Chua, "Analysis of time-varying cellular neural networks for quadratic global optimization," *Int. J. Circuit Theory Appl.*, vol. 26, pp. 109–126, 1998.
- [25] L. O. Chua and L. Yang, "Cellular neural networks: Theory," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1257–1272, Oct. 1988.
- [26] ———, "Cellular neural networks: Applications," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1273–1290, Oct. 1988.
- [27] B. E. Shi and L. O. Chua, "Resistive grid image filtering: Input/output analysis via the CNN framework," *IEEE Trans. Circuits Syst. I*, vol. 39, pp. 531–548, July 1992.
- [28] M. Salerno, F. Sargeni, and V. Bonaiuto, "A dedicated multi-chip programmable system for cellular neural networks," *Proc. Analog Int. Circuits Signal*, vol. 18, pp. 277–288, 1999.
- [29] L. O. Chua and J. M. Cruz, "A CNN chip for connected component detection," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 812–817, July 1991.
- [30] M. Anguita, F. J. Pelajo, A. Prieto, and J. Ortega, "Analog CMOS implementation of a discrete time CNN with programmable cloning templates," *IEEE Trans. Circuits Syst. II*, vol. 40, pp. 215–218, Mar. 1993.
- [31] P. Kinger and M. Steyaert, "A programmable analog cellular neural network CMOS chip for high speed image processing," *IEEE J. Solid-State Circuits*, vol. 30, pp. 235–243, Mar. 1995.
- [32] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84–95, Jan. 1980.
- [33] T. M. Martinetz, S. G. Berkovich, and K. J. Schulten, "Neural gas" network for vector quantization and its application to time-series prediction," *IEEE Trans. Neural Networks*, vol. 4, pp. 558–569, Aug. 1993.
- [34] T. Kohonen, *Self-Organization and Associative Memory*, 3rd ed. New York: Springer Verlag, 1989.
- [35] S. Ridella, S. Rovetta, and R. Zunino, "Plastic algorithm for adaptive vector quantization," *Neural Comput. Appl.*, vol. 7, no. 1, pp. 37–51, Jan. 1998.
- [36] R. L. Baker and R. M. Gray, "Differential vector quantization of achromatic imagery," in *Proc. Int. Picture Coding Symp.*, 1983, pp. 105–106.
- [37] J. Vaisey and A. Gersho, "Variable block-size image coding," in *Proc. ICASSP87*, Apr. 1987, pp. 1051–1054.
- [38] P. Strobach, "Quadtree-structured recursive plane decomposition coding of images," *IEEE Trans. Signal Processing*, vol. 39, pp. 1380–1397, June 1991.

**Sandro Ridella** (M'93) received the Laurea degree in electronic engineering from the University of Genoa, Italy, in 1966.

He is Full Professor at the Department of Biophysical and Electronic Engineering, University of Genoa, Italy, where he teaches Circuits and Algorithms for Signal Processing. In the last eight years his scientific activity has been mainly focused in the field of neural networks.



**Stefano Rovetta** (M'98) received the Laurea degree in electronic engineering (1993) and the Ph.D. degree in models, methods, and tools for electronic and electromagnetic systems (1997), both from the University of Genoa.

He has been working as a Postdoctoral Researcher with the Electronic Systems and Networking Group of the Department of Biophysical and Electronic Engineering, University of Genoa. He is currently with the Department of Computer and Information Sciences of the same University as an Assistant Professor. His research interests include electronic circuits and systems, and neural network theory, implementation and applications.



**Rodolfo Zunino** (S'90–M'90) obtained the Laurea degree in electronic engineering from Genoa University.

From 1986 to 1995 he was a Research Consultant with the Department of Biophysical and Electronic Engineering of Genoa University. He is currently with the same Department as an Associate Professor in Electronics and Industrial Electronics. His main scientific interests include electronic systems for neural networks, distributed control and methods for data representation and processing.