Combined compression and denoising of images using vector quantization

Kannan Panchapakesan, Ali Bilgin, David G. Sheppard, Michael W. Marcellin, Bobby R. Hunt

Department of Electrical and Computer Engineering

The University of Arizona

Tucson, AZ 85721

{kannan,bilgin,sheppard,mwm,hunt}@ece.arizona.edu

http://www-spacl.ece.arizona.edu

ABSTRACT

Compression of a noisy source is usually a two stage problem, involving the operations of estimation (denoising) and quantization. A survey of literature on this problem reveals that for the squared error distortion measure, the best possible compression strategy is to subject the noisy source to an optimal estimator followed by an optimal quantizer for the estimate. What we present in this paper is a simple but sub-optimal vector quantization (VQ) strategy that combines estimation and compression in one efficient step. The idea is to train a VQ on pairs of noisy and clean images. When presented with a noisy image, our VQ-based system estimates the noise variance and then performs joint denoising and compression. Simulations performed on images corrupted by additive, white, Gaussian noise (AWGN) show significant denoising at various bit rates. Results also indicate that our system is robust enough to handle a wide range of noise variances, while designed for a particular noise variance.

Keywords: Denoising, Estimation, Compression, Vector Quantization, Non-linear Interpolative Vector Quantization, Noisy Source Coding.

1. INTRODUCTION

Image compression deals with reducing the amount of information needed to represent an image. Image compression techniques can in general be dichotomized as being lossless or lossy. With lossless compression, the image data can be recovered perfectly but the compression offered is usually moderate. On the other hand, lossy techniques provide excellent compression, but at the expense of loss of image fidelity. Vector quantization, a popular lossy technique, is the quantization of an ordered set of real numbers. It has been used with great success to compress signals such as speech, imagery, and video.

Denoising is essentially the process of estimating the original image data, given a corrupted or noisy version of it. One of the most widely used assumptions for noise is that it is additive, white, and Gaussian. This means that the noise afflicts each pixel in the image on an additive basis, possesses equal power at all frequencies in the spectrum and has a Gaussian shaped probability density function. This paper introduces a joint compression and denoising technique based on non-linear interpolative vector quantization (NLIVQ).[1] The training procedure for the VQ is non-iterative, discrete cosine transform based, and computationally efficient.

2. PROBLEM DESCRIPTION

The problem at hand is to quantize (and thereby compress) a noisy source such that the distortion or error between the output of the quantizer and the clean (original) source is minimized. Consider Figure 1 where X is the clean original signal, N is additive white Gaussian noise (AWGN), Y the noisy version of X, and $Q(Y)$ the quantized version of Y. Mathematically stated, the problem is to minimize

$$E[d(X, Q(Y))],$$

for a given distortion measure d.

Part of the SPIE Conference on Applications of Digital Image Processing XXI • San Diego, California • July 1998
SPIE Vol. 3460 • 0277-786X/98/$10.00

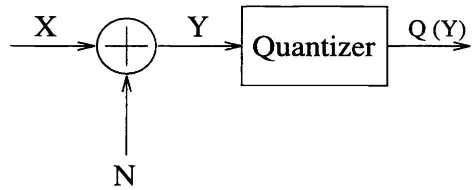

Figure 1. Quantizing a noisy source.

Quantization of a noisy source is a classic problem that has been studied for years. The problem has been considered in various contexts by many, including Dobrushin and Tsybakov,[2] Fine,[3] Sakrison,[4] Wolf and Ziv,[5] Ephraim and Gray,[6] and Ayanoğlu.[7] It has been shown in the literature that the optimal solution to the problem is to cascade the optimal estimator for X (given Y) followed by the optimal quantizer for the estimate. However, elegant as this solution may be, two significant implementational concerns are associated with it. They are as follows:

- The optimal estimator is in general not known or may be extremely complex.
- Realizing the operations of estimation and quantization separately can be computationally inefficient.

These concerns would seem to justify a system design that trades optimality for simplicity and computational efficiency.

One straightforward approach to circumvent the aforementioned problems is to combine the operations of estimation and quantization in one step. This was considered by Rao et. al, who implemented an efficient but sub-optimal VQ system by imposing structural constraints on the encoder while using deterministic annealing to design the VQ.[8] Their simulations were done on data from mathematical models which precludes any comparisons with our approach.

3. NON-LINEAR INTERPOLATIVE VECTOR QUANTIZATION

A Vector Quantizer, Q, with an associated codebook, C, of size K, consisting of k-dimensional vectors, accomplishes the mapping

$$Q : R^k \rightarrow C.$$

The rate of such a quantizer is said to be $r = \frac{log_2 K}{k}$ bits/dimension. The design procedure for such a VQ is usually iterative and involves optimizing the encoder and the decoder at each iteration. Therefore, to design even a moderately high dimensional VQ can be a daunting task.

NLIVQ[1] was introduced by Gersho as a useful complexity reducing technique for quantizing high dimensional vectors. The idea behind NLIVQ, as illustrated in Figure 2, is as follows. Let Z be a random vector of dimension k. As k increases, ordinary full search VQ quickly becomes infeasible. But if it is possible to extract a suitable "feature vector", U, of dimension $n < k$, then Z can be estimated from the vector quantized version, $\hat{U}$, of U. This estimation process of Z from $\hat{U}$ can be accomplished in one step by designing the interpolative decoder such that it is optimal for a given encoder. In this case, the encoder and decoder codebooks would be of the same size but of dimensions n and k respectively. It is of significance to realize here that NLIVQ is a sub-optimal technique, since it does not jointly optimize the VQ encoder and decoder. But, NLIVQ finds its strength in simplicity. It has been used in a wide variety of applications such as joint image restoration and compression,[9] image super-resolution,[10] non-linear speech prediction,[11] lossless predictive image coding,[12] multispectral image compression,[13] and enhancement of transform coding.[14]

4. THE SYSTEM DESIGN STRATEGY

The underlying design strategy for our system closely follows the NLIVQ paradigm discussed above with one important difference, namely, no feature extraction (and hence interpolation) is performed. This means that the encoder and decoder codebooks contain codewords of the same dimension. The design procedure can be described as follows. Let the training set for VQ design be $\{X^i, Y^i\}_{i=1}^N$, where X^i is a clean original image and Y^i the corresponding

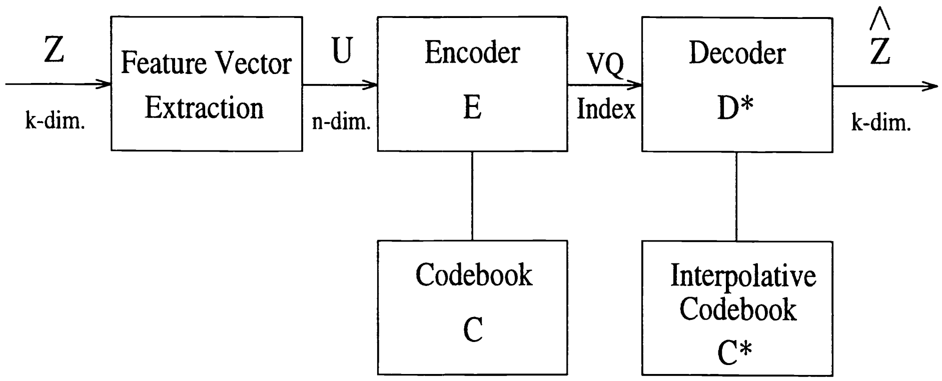

Figure 2. NLIVQ strategy.

noisy image. Decompose each training image into $M \times M$ non-overlapping blocks to be used as the training vectors. Initially, a VQ with encoder, **E**, decoder **D**, and associated encoder codebook C, is designed to minimize the mean-squared quantization error

$$\text{MSE} = E\| y^{ij} - \tilde{y}^{ij} \|^2. \tag{1}$$

The quantized block $\tilde{y}^{ij}$ can be written as

$$\tilde{y}^{ij} = \mathbf{D}(\mathbf{E}(y^{ij})) = \arg\min_{c_l \epsilon C} \| y^{ij} - c_l \|^2, \tag{2}$$

where c_l refers to the lth entry of C.

Now, a new decoder $\mathbf{D}^*$ and its associated codebook C^* is derived by minimizing the conditional expectation

$$E[\| x^{ij} - \tilde{x}^{ij} \|^2 | \mathbf{E}(y^{ij}) = l], \tag{3}$$

where encoder **E** returns the index of the optimal codebook entry. Let $R_l = \{x^{ij} : \mathbf{E}(y^{ij}) = l\}$ for a given set of training blocks. Define entry l of C^* as the centroid of R_l, or

$$c_l^* = \frac{1}{|R_l|} \sum_{x^{ij} \epsilon R_l} x^{ij} \tag{4}$$

where $|R_l|$ is the size of the set R_l. Finally, the denoised image block is given by

$$\tilde{x}^{ij} = \mathbf{D}^*(\mathbf{E}(y^{ij})) = c^*_{\mathbf{E}(y^{ij})}. \tag{5}$$

5. ALGORITHMS FOR CODEBOOK DESIGN

Designing the encoder codebook C is the central issue while the decoder codebook C^* is simply derived from C, as discussed above. The most commonly used algorithm in VQ design is the Lloyd algorithm.[15] But this simple design algorithm is iterative and therefore comes with heavy computational requirements which limits its use to low encoding rates. This motivated us to use a DCT-based, non-iterative technique, described below.[16]

DCT-Based Encoder Design

- Add AWGN of a fixed variance, σ^2, to each of the N clean training images, X^i, to produce the noisy training images, Y^i.
- An image in the training set is first divided into non-overlapping blocks of size $M \times M$. Then the DCT is performed on each of the blocks to produce y^{ij} corresponding to the noisy training images.

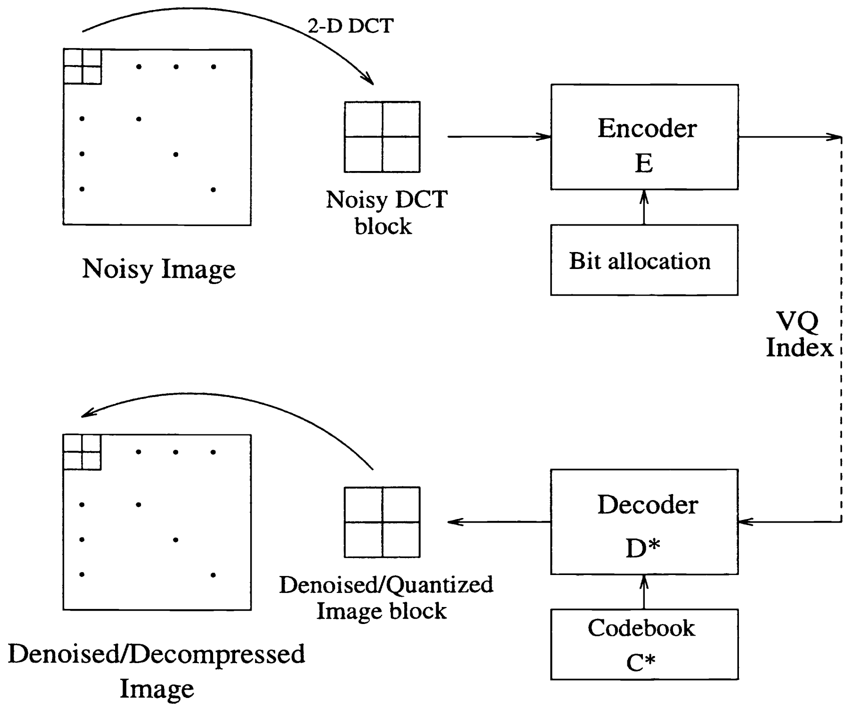

Figure 3. System overview.

- Given a budget of R bits/pixel, allocate the available $L = R \times M \times M$ bits for each block, y^{ij}, such that the quantization error is minimized.
- A pdf-optimized scalar quantizer is designed for each of the M^2 DCT coefficients of y^{ij} according to the rate allocation discussed above. Gaussian and Laplacian distributions are assumed for DC and AC coefficients, respectively.
- Finally, the VQ codeword index from the encoder is generated by concatenating the binary codes from each of the scalar quantizers employed in a given block.

NLIVQ Decoder Design

- Compute the encoder index, $E(y^{ij}) = q$, as defined above, for each **noisy** DCT block, y^{ij}.
- Add the corresponding image block, x^{ij}, from the **clean** training set to the accumulator a_q^* and increment the counter s_q^*.
- Once all of the training blocks have been consumed, each codeword in C^* is computed as the average

$$c_q^* = \frac{1}{s_q^*} a_q^*.$$

Both the encoder and decoder design algorithms are non-iterative and therefore the only computationally intensive part is in computing the block statistics (the mean and variance of each of the DCT coefficients) in order that the bit allocation may be done. The bit allocation at the encoder can be done based on the statistics of either the clean or noisy images. Both of these strategies have been explored in this paper. In practice, the encoder codebook does not need to be stored, since it is but implicitly used. Only the interpolative decoder codebook needs to be stored. With the above algorithms in mind, it would be instructive to look at Figure 3, which provides an overview of the entire system. It is worth noting that there is no inverse DCT in the decoder.

6. SIMULATION RESULTS

Simulations were performed using the algorithms described above, using a training set of 53 grayscale "urban" images, each of size 512×512. All results were obtained from experiments done on a test image outside the training set. There are two parts to this section. The first part deals with the estimation of noise variance in the given image while the second part discusses the joint compression and denoising of this image.

6.1. Noise variance estimation

The noise in a given image is estimated as follows. For a given block size, an estimate of the variance of the highest frequency DCT coefficient, σ_c^2, is first obtained from the set of clean training images. Then, the same is done for the given noisy image, which gives σ_n^2. Since the noise is assumed to be additive and white, an estimate of the noise variance can be obtained by the subtracting σ_c^2 from σ_n^2. The reason the highest frequency DCT coefficient is used is because it is expected to contain the least amount of signal energy and therefore result in the most reliable estimate.

Table 1 shows the noise variance estimation performance using blocks of different sizes. The fact that the variance estimate improves with increasing block size is not surprising because the highest frequency DCT coefficient from the clean images would contain less signal energy, as block size increases. This does not mean that the block size can be indefinitely increased in order to get better estimates. For, doing so would result in unreliable estimates, given a fixed amount of training data. Using the same block size as that used for denoising and compression (discussed below) would be computationally efficient, since in that case there would be no added burden incurred in doing the estimation.

6.2. Combined compression and denoising

The signal-to-noise ratio (SNR), measured always with respect to the clean image, is defined as

$$\text{SNR} = 10 \log_{10} \frac{\text{signal power}}{\text{noise power}} \text{ dB.}$$

Training blocks of size 2×2 were employed which resulted in 4-dimensional training vectors, with bits being allocated to each of the DCT coefficients depending on the desired overall bit rate. For example, to realize a bit rate of 2 bpp, a total of 8 bits were distributed among the 4 DCT coefficients.

Table 2 displays results for the test image corrupted with an AWGN of variance 400. This noise variance equates to an SNR of 17.02 dB. VQ $_{noisy}$ is a vector quantizer designed to minimize quantization error without any explicit attempt to incorporate denoising. This quantizer was trained using noisy images at both encoder and decoder. This system does not result in any significant denoising. The next column shows the performance of VQ $_{clean}$, which was similarly trained with clean images at encoder and decoder. When this VQ is used to compress the noisy test image, it effects moderate denoising but its performance is limited by the fact that it has no knowledge of the noise.

Table 1. Noise variance estimation performance.

Actual noise variance	Estimated noise variance using a block size of		
	2×2	3×3	4×4
200	229	214	208
400	431	416	398
800	842	821	807

Table 2. SNR(dB) of decompressed images at different rates (bits/pixel). SNR of noisy image = 17.02 dB.

R	VQ $_{noisy}$	VQ $_{clean}$	NLIVQ $_{NA}$	NLIVQ $_{CA}$
2.0	17.02	18.37	19.01	19.45
1.0	16.70	18.22	17.43	18.30

Table 3. SNR(dB) at 2 bits/pixel from a system designed for a noise variance of 400 but tested on noise variances of 200, 400, and 800, respectively.

Noise variance	Noisy image	NLIVQ $_{CA}$ image
200	20.01	20.65
400	17.02	19.45
800	13.98	17.83

Table 4. SNR(dB) at 2 bits/pixel from a system designed for and tested on noise variances of 200, 400, and 800, respectively.

Noise variance	Noisy image	NLIVQ $_{CA}$ image
200	20.01	20.98
400	17.02	19.45
800	13.98	17.90

The columns labeled NLIVQ $_{NA}$ and NLIVQ $_{CA}$ are non-linear interpolative vector quantizers designed as explained in the previous section (using noisy and clean images at the encoder and decoder, respectively) with the bit allocations at the encoder based on noisy and clean image characteristics, respectively. That NLIVQ $_{CA}$ outperforms NLIVQ $_{NA}$ is intuitively reasonable. In the latter case, the noise biases the bit allocation procedure to artificially emphasize high frequencies.

At a rate of 1.0 bpp, it may be noticed that VQ_{clean} actually provides a better SNR value than NLIVQ $_{NA}$. This can be attributed to the fact that the quantizers designed in these simulations are only locally optimal. Overall, the NLIVQ $_{CA}$ system performs the best, providing SNR gains (over the noisy uncompressed image) of 2.43 dB and 1.28 dB at rates 2.0 bpp and 1.0 bpp, respectively.

In order to investigate the robustness of the NLIVQ $_{CA}$ system, the following simulations were performed. First, the NLIVQ $_{CA}$ system was designed for a noise variance of 400 at 2 bpp. Then, this system was used on the test image corrupted by noise of variances 200, 400, and 800 (corresponding to SNRs of 20.01 dB, 17.02 dB, and 13.98 dB), respectively. The results are shown in Table 3.

Next, an NLIVQ $_{CA}$ system was designed for each of the above noise variances, and used on the test image corrupted by noise with the same variance. These results are given in Table 4. To quantify the loss in not using an appropriately designed system, we compare the results in Tables 3 and 4. It can be seen that for a noise variance of 200, only a 0.33 dB loss is incurred, while for a noise variance of 800, this loss is as little as 0.07 dB. This indicates that the system is robust and is capable of handling different noise variances than it was designed for.

The performance in estimating the noise variance as well as joint compression and denoising suggests that the existing system could be used in a practical scenario as follows. Given a noisy image, the noise variance is first estimated. Then a suitable quantizer is chosen from a set of pre-designed VQs (for various noise variances in a useful range) and applied on the image to achieve the desired compression and denoising.

ACKNOWLEDGMENTS

This work was supported in part by the United States Air Force, the Air Force Office of Scientific Research (Grant No. F49620-97-1-0099), and the National Science Foundation (Grant No. NCR-9258374).

REFERENCES

1. A. Gersho, "Optimal non-linear interpolative vector quantization," *IEEE Trans. Commun.* **38**, pp. 1285–87, September 1990.
2. R. L. Dobrushin and B. S. Tsybakov, "Information transmission with additional noise," *IEEE Trans. on Inform. Theory* **8**, pp. 293–304, 1962.

3. T. Fine, "Optimum mean-square quantization of a noisy input," *IEEE Trans. on Inform. Thoery* **11**, pp. 293–294, April 1965.
4. D. Sakrison, "Source encoding in the presence of random disturbance," *IEEE Trans. on Inform. Theory* **14**, pp. 165–166, January 1968.
5. J. K. Wolf and J. Ziv, "Transmission of noisy information to a noisy receiver with minimum distortion," *IEEE Trans. on Inform. Theory* **16**, pp. 406–411, July 1970.
6. Y. Ephraim and R. M. Gray, "A unified approach for encoding clean and noisy sources by means of waveform and autoregressive model vector quantization," *IEEE Trans. on Inform. Theory.* **34**, pp. 826–834, July 1988.
7. E. Ayanoğlu, "On optimal quantization of noisy sources," *IEEE Trans. on Inform. Theory* **36**, pp. 1450–1452, November 1990.
8. A. Rao, D. Miller, K. Rose, and A. Gersho, "A generalized VQ method for combined compression and estimation," *Proc. IEEE Intl. Conf. Acoustics, Speech, and Sig. Proc.* , pp. 2032–2035, 1996.
9. K. Panchapakesan, A. Bilgin, M. W. Marcellin, and B. R. Hunt, "Joint compression and restoration of images using wavelets and non-linear interpolative vector quantization," *Proc. ICASSP* **5**, pp. 2649–52, May 1998.
10. D. G. Sheppard, K. Panchapakesan, A. Bilgin, B. R. Hunt, and M. W. Marcellin, "Lapped nonlinear interpolative vector quantization and image super-resolution," *Proc. of Thirty First Asilomar Conference on Signals and Computers* , November 1997.
11. S. Wang, E. Paksoy, and A. Gersho, "Performance of non-linear prediction of speech," *Proc. of Intl. Conf. on Spok. Lang. Proc.* **1**, pp. 29–32, 1990.
12. M. J. Slyz and D. L. Neuhoff, "A non-linear VQ-based predictive lossless image coder," *Proc. IEEE Data Comp. Conf.* , pp. 304–310, 1994.
13. S. Gupta and A. Gerso, "Feature predictive vector quantization of multispectral images," *IEEE Trans. on Geos. and Rem. Sens.* **30**, pp. 491–501, 1992.
14. S. W. Wu and A. Gersho, "Enhancement of transform coding by nonlinear interpolation," *Proc. of SPIE Vis. Comm. and Image Proc.* , pp. 487–498, 1991.
15. Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design.," *IEEE Trans. Commun.* **28**, pp. 84–95, January 1980.
16. D. G. Sheppard, A. Bilgin, M. S. Nadar, B. R. Hunt, and M. W. Marcellin, "A vector quantizer for image restoration," *IEEE Trans. Image Proc.* **7**, pp. 119–124, January 1998.