EXTENDED CLUSTER INFORMATION VECTOR QUANTIZATION (ECI-VQ) FOR ROBUST CLASSIFICATION

Jon A. Arrowood ⋆ *and Mark A. Clements*

Center for Signal & Image Processing
Georgia Institute of Technology, Atlanta, USA
{jon,clements}@ece.gatech.edu

ABSTRACT

This paper presents a novel extension to vector quantization referred to as *Extended Cluster Information* (ECI). In this method the decoder retains more general statistics about the VQ clusters found during codebook training than the single prototypical point of conventional VQ systems. Typically this information is unnecessary, however if the items being compressed are feature space vectors used as input to a statistical pattern classification system, the extra probabilistic information can be used during the classification as in Bayes Predictive Classification (BPC) to improve recognition results. To demonstrate ECI-VQ, a simple experiment is described where the Aurora2 distributed speech recognition front end is altered to provide more aggressive Mel Frequency Cepstral Coefficient (MFCC) compression. As the bit-rate drops, the corresponding recognition performance suffers. It is then shown that using ECI-VQ as the input to an Uncertain Observation (UO) speech recognizer, a number of errors due to compression can be corrected with no extra cost in bit-rate.

1. INTRODUCTION

Standard vector quantization (VQ) achieves fantastic compression rates for many scenarios. This is accomplished by breaking the feature space for the problem into a finite number of regions, and replacing individual, continuously-valued samples in this space by a single prototypical value representing the region containing the sample. For most uses, the hard decision implied in VQ is appropriate. However, if the VQ step is the front end to a pattern classification system, this hard decision represents an unnecessary loss of information about the original sample. This paper describes a new extension to vector quantization referred to as the *Extended Cluster Information* (ECI) method, that augments VQ output to probabilistically represent each cluster. This extra information is embodied as a probability density function, giving the likelihood at the decoder of any point in the feature space, given that some codebook cluster index i was transmitted. This PDF can be determined at the time of training of the VQ codebook, just as the standard single prototypical point for each cluster is usually determined. This PDF is then embedded into the design of the VQ decoder. Transmission still consists of only a cluster index, but the decoder can now give full cluster statistics as output.

To leverage this extra cluster information, the classifier needs to have the ability to handle observations that are represented probabilistically. Such a method exists, and is often referred to as a Bayes Predictive Classifier (BPC) [1]. When a classifier has been trained using undistorted, non-compressed features, BPC can integrate observation uncertainty into the classification process in an optimal manner.

The new technique will be demonstrated in this paper for the specific case of distributed speech recognition (DSR). DSR often involves the VQ of MFCC feature vectors for transmission to a remote speech recognition server. When the VQ compression is aggressive, it will degrade recognition performance. It will be shown that replacing standard VQ with ECI-VQ, and using an Uncertain Observation (UO) decoder in place of a standard decoder, a significant amount of performance lost due to compression can be regained.

2. EXTENDED CLUSTER INFORMATION VECTOR QUANTIZATION (ECI-VQ)

Training of a vector quantization system begins as shown in the two-dimensional example in Figure 1(a). A large amount of training data is separated into some finite number of clusters, three in this case, ω_+, ω_x, and ω_o. Feature space is thus separated into $I = 3$ disjoint regions.

Standard VQ classification takes an input vector x, as shown in Figure 1(b) and based upon the region where the sample resides, assigns the sample to that cluster. For this example, assignment is performed by calculating the distance d_x, d_o, and d_+ to each of the clusters, and assigning x to the nearest, ω_+ in this case.

For transmission, all that needs to be sent from the VQ

⋆ now with Nexidia, Inc., in Atlanta, GA

coder to the VQ decoder is the index of the cluster. The standard VQ decoder will then replace the index with a single prototypical point representing the cluster. For this example, this prototypical point is the mean of the training data assigned to the cluster, μ_+, as shown in Figure 1(c).

This single prototypical point μ_+ was found during VQ codebook training, and stored in the decoder. Note that there is no reason that further information about the cluster could not also be stored in the decoder. Instead of making a hard decision and choosing only a single point, the cluster can be represented probabilistically, giving the likelihood of any point in feature space, conditioned on receiving a cluster index. This is shown graphically in Figure 1(d), giving as VQ decoder output a Gaussian PDF with mean μ_+ and covariance σ_+.

Several choices exist for how to represent the cluster PDFs. Gaussian is a good choice, although as it has infinite tails, it will give non-zero probabilities for points that would have been assigned to other VQ clusters by the coder. Another possibility is a uniform PDF over the region of feature space represented by the cluster. This is appropriate only if all clusters represent finite regions.

A comparison of standard VQ and ECI-VQ flowcharts are shown in Figure 2(a) and (b), respectively. The two systems perform classification in an identical manner, and transmit the same information over the channel. The only VQ change is to the decoder, with ECI-VQ giving extra output information (in this figure, Gaussian likelihoods) learned during VQ codebook training, which can then be used in the statistical classifier.

No matter which PDF is chosen, the final result is that the bit-rate of the VQ system remains unchanged, but the decoder is allowed to make soft decisions rather than the conventional hard decision. The next step is to identify a pattern classification method that allows observation uncertainty.

It is worthwhile to briefly compare this method to fuzzy-VQ, which allows the input sample to be described by several VQ clusters, rather than only using a single cluster. While this method also allows uncertainty about the classification to be passed to the decoder, there is a corresponding increase in datarate, as all cluster indices must be transmitted. For distributed recognition, this increased datarate defeats the purpose of acheiving a low bitrate, thus fuzzy VQ is not applicable to this problem. For systems that do not have to transmit the VQ indices, of course, fuzzy VQ remains a viable technique.

3. REVIEW OF UNCERTAIN OBSERVATION DECODING

An interesting consequence of the extended cluster information scheme described above is that it allows for arbitrary

(a) **VQ Training:** Training samples are grouped into some number of clusters.

(b) **VQ Classification:** Input sample x is assigned to the nearest cluster (here, $x \in \omega_+$).

(c) **Standard VQ Output:** A single point representative of x's cluster (μ_+ in this case).

(d) **ECI-VQ Output:** The PDF describing the likelihood $\Pr[x|\omega_+]$ (here, represented by mean μ_+ and covariance Σ_+).

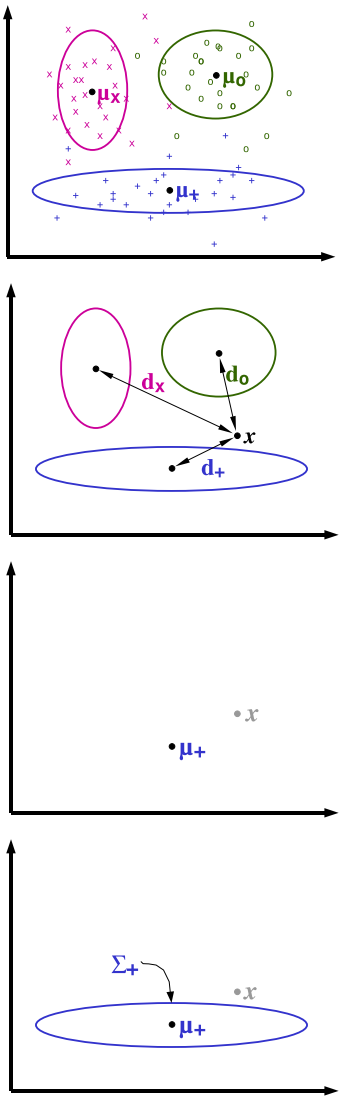

Fig. 1: Visualization of VQ training, VQ classification, standard VQ output, and ECI-VQ output. Training and classification (steps (a) and (b)) are the same for both standard VQ output (shown in (c)) and ECI-VQ output (shown in (d)).

PDF descriptions to be generated as output from the VQ decoder. As several techniques have recently been developed to use PDF descriptions of features for robust recognition in place of the standard points in feature space [2, 3, 4], ECI-VQ is ideal for performing robust speech recognition decoding.

For this paper, ECI-VQ was used as a front-end to the Uncertain Observation HMM decoding algorithm described in [2]. Instead of calculating the state j output probability, $b_j(\mathbf{x})$, for a frame of speech by finding the probability of a single point, $\mathbf{x}$, in space representing that speech frame, the UO decoding algorithm finds the probability of all pos-

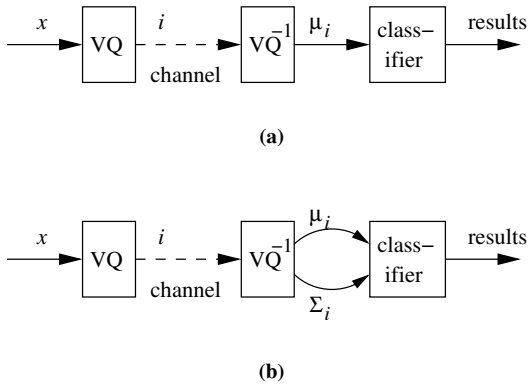

Fig. 2: Flowchart of (a) standard VQ, and (b) ECI-VQ with Gaussian likelihoods.

sible observations, weighted by their respective likelihoods. Thus, the state output probability calculation is in general specified as

$$\Pr[\mathbf{y}_n|q_t=j,\mathcal{W}_n] = \int_{-\infty}^{\infty} f_n(\boldsymbol{\vartheta}) b_j(\boldsymbol{\vartheta})\, d\boldsymbol{\vartheta} \qquad (1)$$

where $f_n(\mathbf{x})$ is some PDF describing $\Pr[\mathbf{x}|\mathbf{y}_n,\mathcal{W}_n]$, the likelihood of unobserved clean speech feature vector $\mathbf{x}_n$ being $\mathbf{x}$ given noisy observation $\mathbf{y}_n$ and distortion model $\mathcal{W}_n$.

For the particular case of a K mixture Gaussian speech model and a single Gaussian speech observation PDF, the likelihood distributions are

$$b_j(\mathbf{x}) \sim \sum_{k=1}^{K} \mathcal{N}(\boldsymbol{\mu}_{jk}, \boldsymbol{\Sigma}_{jk}) \qquad (2)$$

$$f_n(\mathbf{x}) \sim \mathcal{N}(\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n) \qquad (3)$$

and the state output probability calculation for the UO decoding algorithm given in Equation 1 simplifies to [2]:

$$\sum_{k=1}^{K} c_{jk}\, \mathcal{N}(\boldsymbol{\mu}_{jk}, \boldsymbol{\Sigma}_{jk} + \boldsymbol{\Sigma}_n)\big|_{\boldsymbol{\mu}_n} \qquad (4)$$

An alternate implementation of particular interest for ECI-VQ is the use of a uniform random variable representation for the observation. In this case, the state output likelihood given in Equation 1 is:

$$\Pr\left[\mathbf{y}_t|q_t=j,\mathcal{W}_n\right] = A_i \int_{R_i} b_j(\boldsymbol{\vartheta})\, d\boldsymbol{\vartheta} \qquad (5)$$

Cluster regions for VQ are almost certain to be non-rectangular, and finding the boundaries for the integral is not necessarily straightforward. Thus for this paper, only Gaussian observations were considered.

4. DEMONSTRATION EXPERIMENT

To demonstrate ECI-VQ, experiments using the Aurora2 distributed speech recognition framework were used. Three changes were made to the baseline Aurora2 recognition experiment. First, as the goal was to measure the drop in performance due to MFCC feature compression, and not background noise, only clean speech was necessary for training and testing. Training thus used all of the clean-training data, and testing used only the Test A clean1 speech subset.

Second, the system was altered to only use the 13-dimensional static MFCC features, instead of the usual 39-dimensions. This lowered the baseline performance, but allowed the experiments to better focus on the effect of the ECI-VQ method.

The third change was to alter the vector quantizer. The baseline ETSI front end for DSR uses a split VQ quantizer with a bit-rate of 4.8kbps, and gives no significant change in performance compared to uncompressed features on clean speech. ECI-VQ is designed for systems that have compression aggressive enough to cause classification errors, thus the VQ step was altered to provide more compression. The exact design of the VQ classifier was not particularly the issue, only that the VQ be aggressive, thus rather than retraining a full new system, a low bit-rate system was approximated by altering the original ETSI quantizer. The original system operated as seven independent streams, each operating on two cepstral or energy coefficients, using between 5 and 8 bits per stream. To generate a more aggressive quantizer, each stream was reduced by two bits, by randomly discarding codebook entries. This resulted in a new, more aggressive VQ system with a bit-rate of 3.3kbps. This new system showed a significant drop in recognition accuracy due to the feature vector compression.

To implement ECI-VQ as described in Section 2, the codebook training data is required in order to collect cluster statistics. Since new codebooks were not trained, but derived from an already-existing VQ system, the actual training data was not available. An alternate approach was taken to get around this problem. The corpus of training data was passed through the VQ system, and cluster indexes collected for each frame. This allowed clusters to be generated for each codebook index, approximating what would have been found had the actual VQ training data been available.

For each codebook index, statistics about the clusters obtained using the training data were embedded into the VQ decoder. The cluster for each codebook entry was represented by a single-mixture Gaussian random variable with diagonal covariance matrix. This allowed ECI-VQ to generate probabilistic features for an Uncertain Observation speech decoder that used Equation 4 for the state output probability calculations. The next section describes the results found when comparing the standard recognition method to

the ECI-VQ method.

5. RESULTS

The baseline system was trained using speech that had not been compressed. This was used to test two scenarios: uncompressed clean speech, and speech compressed with the 3.3kbps VQ systems described above. Decoding operated using only the single prototypical point in feature space given by the standard VQ decoder. The results, shown in the first two lines of Table 1, show that aggressive VQ causes a significant drop in recognition performance. Although not shown in the table, using the standard 4.8kbps VQ did not adversely affect recognition from the non-VQ results.

A second baseline system was designed to model speech and VQ distortion jointly, with speech models trained using quantized feature vectors. Results from this method, in the third line of Table 1 showed a tremendous drop in performance, giving results worse than those found testing compressed data against models trained on uncompressed feature data. One hypothesis for why seemingly mismatched conditions perform better than this matched scenario is that there is not enough training data to jointly model the two systems, and more training data would fix this issue. A second hypothesis is that the speech models in the recognition system are using continuous density mixtures, which does not correspond well to training with quantized features.

The final system tested uses speech models trained with uncompressed speech, but testing uses a UO decoder receiving single mixture Gaussian probabilistic features from an ECI-VQ decoder. Results, given in the fourth row of Table 1, show that over one third of the errors induced by aggressive VQ can be removed. For example, in the 3.3kbps case, the standard method gives an error rate of 3.6%, 1.5% above the 2.1% of the baseline system with no compression. ECI-VQ gives an error rate of 3.0%, only 0.9% above the baseline, a 40% reduction in VQ-induced errors.

Table 1: Recognition results demonstrating ECI-VQ

training data	testing data	digit error rate
no VQ	no VQ	2.1%
no VQ	VQ	3.6%
VQ	VQ	5.6%
no VQ	ECI-VQ	3.0%

The experiments in this section used only clean data in order to focus on ECI-VQ. An interesting side note, however, is that compressed *noisy* speech features show a drop in recognition accuracy beyond that of uncompressed noisy speech [5]. Noisy feature vectors are less well represented in the VQ training data, resulting in compression using clusters with higher variances. The explicit use of cluster variances to this scenario is thus an interesting future direction.

6. CONCLUSIONS

This paper has presented a new paradigm for extending the output of vector quantization, called ECI-VQ for *Extended Cluster Information VQ*. After classifying an input sample using the VQ coder, the cluster index is transmitted as usual, however instead of the VQ decoder giving a single prototypical point, the output is the likelihood of any point in the feature space given the cluster index. Such a method is useful when vector quantization is used in front of a statistical classifier.

As an example, ECI-VQ was applied to a distributed speech recognition task that uses quantized feature vectors. By extending the speech recognizer to use probabilistic input features that come out of ECI-VQ, the loss in recognition performance due to feature compression was cut nearly in half, compared to using the output of standard VQ.

The speech recognition example showed an interesting example of using training data to find cluster statistics on an existing VQ system. The only constraint is that codebook index that is transmitted must be visible, and thus pre existing VQ systems can be adapted to give ECI-VQ output.

The net result of using ECI-VQ in front of a classifier, such as the speech recognition example given in this paper, is that it allows for the underlying data (ie, speech) to be modeled separately from the distortion caused by quantization. While these two can be modeled jointly, an independent treatment is potentially beneficial.

7. REFERENCES

[1] B. D. Ripley and N. L. Hjort, *Pattern Recognition and Neural Networks*, Cambridge University Press, 1996.

[2] J. A. Arrowood and M. A. Clements, "Using observation uncertainty in HMM decoding," in *Proc. ICSLP*, 2002, pp. 1561–1564.

[3] N. Yoma and M. Villar, "Speaker verification in noise using a stochastic version of the weighted Viterbi algorithm," *IEEE Trans. Speech and Audio Proc.*, vol. 10, pp. 158–166, 2002.

[4] L. Deng, J. Droppo, and A. Acero, "Exploiting variances in robust feature extraction based on a parametric model of speech distortion," in *Proc. ICSLP*, 2002, pp. 2449–2452.

[5] N. Parihar, J. Picone, D. Pearce, and H. G. Hirsch, "Performance analysis of the aurora large vocabulary baseline system," in *Proc. Eurospeech '03*, Geneva, Switzerland, 2003.