Use of multiple vector quantisation for semicontinuous-HMM speech recognition

A.M. Peinado
J.C. Segura
A.J. Rubio
V.E. Sánchez
P. García

Indexing terms: Error rate, Hidden Markov models, Probability density function, Speech modelling, Speech recognition

Abstract: Although the continuous hidden Markov model (CHMM) technique seems to be the most flexible and complete tool for speech modelling, it is not always used for the implementation of speech recognition systems because of several problems related to training and computational complexity. Thus, other simpler types of HMMs, such as discrete (DHMM) or semicontinuous (SCHMM) models, are commonly utilised with very acceptable results. Also, the superiority of continuous models over these types of HMMs is not clear. The authors' group has recently introduced the multiple vector quantisation (MVQ) technique, the main feature of which is the use of one separated VQ codebook for each recognition unit. The MVQ technique applied to DHMM models generates a new HMM modelling (basic MVQ models) that allows incorporation into the recognition dynamics of the input sequence information wasted by the discrete models in the VQ process. The authors propose a new variant of HMM models that arises from the idea of applying MVQ to SCHMM models. These are SCMVQ-HMM (semicontinuous multiple vector quantisation HMM) models that use one VQ codebook per recognition unit and several quantisation candidates for each input vector. It is shown that SCMVQ modelling is formally the closest one to CHMM, although requiring even less computation than SCHMMs. After studying several implementation issues of the MVQ technique, such as which type of probability density function should be used, the authors show the superiority of SCMVQ models over other types of HMM models such as DHMMs, SCHMMs or the basic MVQs.

1 Introduction

During the last few years, hidden Markov models (HMM) have been successfully applied to acoustic modelling for speech recognition. Two main variations of HMM have been widely used: discrete HMM (DHMM) and continuous HMM (CHMM). The first use nonparametric discrete output probability distributions, due to a previous VQ process. CHMMs use parametric densities to model the output probabilities, on the assumption that the observed signals have been generated by a mixed Gaussian process or an autoregressive process [1]. The main problem with DHMMs is the loss of information about the input signal during the VQ process. CHMMs avoid this problem by using probability density functions (PDFs). Thus, CHMM modelling appears to be a more flexible and complete tool for speech modelling. In spite of this, such models are not always used for the implementation of speech recognition systems. There are several reasons for this. The main problem is the large number of parameters to estimate. In order to obtain good estimates, a large amount of computation and a large database are required. These requirements cannot always be satisfied with the available resources. These are strong restrictions that may make the use of DHMM [2] more attractive.

In order to avoid such problems of continuous modelling, Huang *et al.* [2] propose the use of semicontinuous HMM (SCHMM) models, a hybrid modelling that uses several VQ candidates instead of only the best one, as in DHMMs. Huang has shown that SCHMMs can achieve better results than CHMMs. Our group has recently proposed a new approach based on the use of multiple vector quantisation (MVQ) for HMMs. The resultant new modelling has been called MVQ-HMM (or simply MVQ) models [3]. A MVQ model is composed of a VQ codebook and a discrete HMM. These new models have been introduced as a direct way to incorporate to the system dynamics the information lost in the VQ process when using the discrete approach. In order to do this, each MVQ model uses its own VQ codebook to evaluate the average distortion of the input utterance. With the same amount of computation, the MVQ modelling can clearly outperform DHMMs and achieve similar or better results than SCHMMs (with less computation). Other advantage of MVQ modelling is the possibilities of applying discriminative VQ training due to the use of one specific codebook per recognition unit [4].

In this paper, we study several implementation issues of MVQ models, such as the selection of the probability density functions to model the representation space, and the composition of the distortion and sequential information for minimum error rate. From these preliminary studies, we propose a new variant of HMM modelling based on the application of the MVQ techique to semicontinuous modelling. The new approach will be

Paper 1576K (E5), first received 1st June and in revised form 13th September 1994

The authors are with Dpto. de Electrónica y Tecnología de Computadores, Facultad de Ciencias, Universidad de Granada, 18071-Granada, Spain

called SCMVQ-HMM modelling (semicontinuous HMMs with multiple vector quantisation) or simply SCMVQs. These new models can be also considered as MVQHMM models using several quantisation candidates. We show that SCMVQs can obtain the best recognition performance in comparison to DHMMs, SCHMMs and MVQs. Furthermore, we also show that if the PDF utilised for SCMVQs is applied to SCHMMs, this improves their performance.

2 Generalised HMM framework

The difference between the different HMM variants is in the computation of the output probabilities $b_i(\boldsymbol{x})$ of a vector $\boldsymbol{x}$ in state s_i, given a model λ. The most general form corresponds to CHMM modelling, where $b_i(\boldsymbol{x})$ is modelled by a mixture of PDFs of the form

$$b_i(\boldsymbol{x}) = P(\boldsymbol{x} \mid s_i, \lambda) = \sum_{v_k \in V(s_i, \lambda)} P(\boldsymbol{x} \mid v_k, s_i, \lambda) P(v_k \mid s_i, \lambda) \quad (1)$$

where each $P(\boldsymbol{x} \mid v_k, s_i, \lambda)$ is a log-concave or elliptically symmetric density [1] (Gaussian throughout this work) with mean vector $\boldsymbol{y}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$, and $\boldsymbol{x}$ is the input vector. Each PDF is labelled by one symbol v_k ($k = 1, \ldots, M$) that belongs to the set $V(s_i, \lambda)$ defined for state s_i in model λ. Factors $P(v_k \mid s_i, \lambda)$ are the mixture coefficients (their sum extended over $V(s_i, \lambda)$ must be one).

A first simplification can be made by forcing all the states to share the same set of PDFs, $V(s_i, \lambda) = V\ (\forall s_i, \lambda)$, which leads to the semicontinuous HMM approach. The output probabilities are now computed as

$$P(\boldsymbol{x} \mid s_i, \lambda) = \sum_{v_k \in V} P(\boldsymbol{x} \mid v_k) b_i(v_k) \quad (2)$$

The set V can be obtained from the construction of a VQ codebook. The sum of eqn. 2 is usually reduced only to the best set of candidates (the most probable ones).

The DHMM approach is easily extracted from SCHMM modelling considering only the best VQ candidate (the nearest VQ centre)

$$P(\boldsymbol{x} \mid s_i, \lambda) = P(\boldsymbol{x} \mid o) P(o \mid s_i, \lambda) \quad (3a)$$

$$o = \min_{v_k \in V}{}^{-1}[d(\boldsymbol{x}, \boldsymbol{y}_k)] \quad (3b)$$

where $d(\boldsymbol{x}, \boldsymbol{y})$ is the utilised distortion measure. In this case, it can be easily derived that the probability of an input sequence $X = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$ (with $O = o_1, \ldots, o_T$ as a quantised version) in model λ is

$$P(X \mid \lambda) = P(X \mid O) P(O \mid \lambda) \quad (4)$$

Thus, only $P(O \mid \lambda)$ (probability of generation) is useful for recognition purposes, since $P(X \mid O)$ (probability of quantisation) does not depend on the considered model λ.

The basic MVQ-HMM modelling is based on the use of one codebook per model $V(s_i, \lambda) = V(\lambda)$ for all s_i in model λ and on the assumption of nonoverlapped PDFs. Thus, the output probabilities can be expressed as

$$b_i(\boldsymbol{x}) = P(\boldsymbol{x} \mid o, \lambda) b_i(o) \quad (5a)$$

$$o = \max_{v_k \in V(\lambda)}{}^{-1}[P(\boldsymbol{x} \mid v_k, \lambda)] \quad (5b)$$

In this case, it is also possible to obtain a decomposition of the probability of a sequence X

$$\log P(X \mid \lambda) = \log P(X \mid O, \lambda) + \log P(O \mid \lambda) \quad (6)$$

The probability of generation can be estimated in the same way as for DHMM models (each model using its own VQ codebook). The main difference between DHMM and MVQ models is that the probability of quantisation cannot be removed now, since it is different for each model. Thus, it can be considered that a MVQ model is composed of a VQ codebook and a discrete HMM, each of them providing its own score (quantisation and generation scores).

If we consider that the parameter set of a MVQ model can be decomposed as $\lambda = (\theta, \phi)$, where θ is the parameter set of PDFs $P(\boldsymbol{x} \mid v_j, \lambda)$ (related to the VQ codebook), and $\phi = (A, B, \Pi)$ (related to the associated discrete HMM model), one possible training method [3] consists of an independent estimation of parameters θ and ϕ. The first ones can be estimated by applying the LBG algorithm [5], and the second ones by means of the Baum–Welch algorithm [6] (once the quantised versions O of the training sequences X have been obtained from the VQ codebook trained in the first step). We will prove later that this method is indeed a ML estimation.

3 Implementation of a MVQ-based system

Once MVQ modelling is defined, some restrictions must be imposed in order to improve system performance. In this section, we explore several types of diagonal covariance matrices and the effect of modifying the weights of quantisation and generation probabilities in (6).

For simplicity, the different techniques introduced in this paper are tested and tuned on an isolated word recognition system (due to the large number of experiments performed). The vocabulary is made up of 16 words, the ten Spanish digits and six keywords, uttered three times by 20 male and 20 female speakers, i.e. 1920 different signals in the database. The average SNR measured over this database is 24 dB, which corresponds to the environment of a workroom with computer noise. In order to increase the statistical significance of the results, the speakers are separated into five disjoint groups containing utterances from eight different speakers (four male, four female), to be utilised for test (the rest for training). The result is the realisation of five different speaker-independent experiments, whose error results are finally averaged. Thus, the whole database (1920 utterances) is used to estimate the error rate. This procedure is similar to the well-known leaving-one-out technique for error probability estimation. The data were analysed using 32 ms frames, overlapped 16 ms, applying 10-order LPC analysis. The feature vectors incorporate 14 liftered cepstrum coefficients, 14 delta cepstrum coefficients and delta energy (delta features are computed with ± 3 frame intervals), and are compared with a multifeature weighted distance measure similar to that described in Reference 7. The HMM topology is left-to-right with ten states. No duration modelling is used (unlike in Reference 3).

3.1 Selection of covariance matrices

Each PDF (labelled with $v_k \in V(\lambda)$) used for MVQ modelling is assumed to be a multivariate Gaussian density, with a mean vector $\boldsymbol{y}_k$ (VQ centre) and a diagonal covariance matrix $\boldsymbol{\Sigma}_k$.

We have tested three different forms for the covariance matrices in three different experiments:

EXP1: Using a different covariance matrix for each *pdf* v_k. Thus, the elements of the main diagonal $\{\sigma_{ki}^2,\ i = 1, \ldots, p\}$ are specific for the cell of centre $\boldsymbol{y}_k$.

EXP2: Using only one covariance matrix Σ_λ shared by all the PDFs in the codebook of model λ, where each element of the diagonal is the average distortion of the corresponding feature in that codebook.

EXP3: Using only one covariance matrix $\Sigma_\lambda = \sigma_\lambda^2 I$ for all the PDFs in the codebook of λ, where σ_λ^2 is the average distortion per feature in the codebook, and I is the identity matrix. Each PDF can be written as

$$P(x \mid v_k, \lambda) = (2\pi\sigma_\lambda^2)^{-p/2} \exp\left\{-\frac{1}{2\sigma_\lambda^2}\|x - y_k\|^2\right\} \quad (7)$$

As we can see, there is a linear relation between the logarithm of eqn. 7 and the distance $\|x - y_k\|^2$.

Fig. 1 shows the results of these three experiments, using 4, 8, 16 and 32 centres per codebook. In general, EXP2

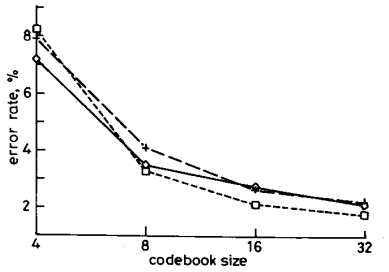

Fig. 1 *Error rate versus codebook size for EXP1 (◇), EXP2 (+) and EXP3 (□)*

obtains the worst results. Although EXP1 provides the best result for four centres, EXP3 presents a better behaviour for eight or more centres. We find two possible reasons for this. First, EXP3 is the only experiment for which the probability measure is coherent with the VQ distance used in this work, that is, the nearest centre of an input vector also correponds to the most probable PDF. Second, EXP3 uses only one parameter, σ_λ^2, to represent all the covariance matrices of all the PDFs in the codebook, which implies a great reduction in the number of parameters to train, lightening the problem of insufficient training. Therefore, the PDF given by eqn. 7 will be adopted for MVQ models.

3.2 Composition of probabilities

We can see from eqns. 6 and 7 that the purpose of the MVQ modelling described above is to add to the log-score provided by the discrete HMM model (probability of generation) a new score (probability of quantisation) that is linearly related to the average distortion of the input sequence X in the codebook of model λ. The idea of recognising without time alignment using several VQ codebooks has been already proposed and successfully applied by Burton *et al.* [8].

It must be pointed out that the composition of probabilities in eqn. 6 may not be optimal because of the assumption of diagonal Gaussian PDFs for the quantisation process. The optimisation of the composition can be achieved experimentally, introducing a weighting factor α

$$\log P(X \mid \lambda) = \alpha \log P(X \mid O, \lambda) + \log P(O \mid \lambda) \quad (8)$$

with $\alpha = \mu/(1 - \mu)$, where μ is a composition factor that takes values from 0 (only probability of generation) to 1 (only probability of quantisation). Fig. 2 shows how the error rate varies as a function of μ, in the range 0.25–0.75, for 8, 16 and 32 centres per codebook. Although $\mu = 0.5$ ($\alpha = 1$) is not a bad selection, there is a minimum

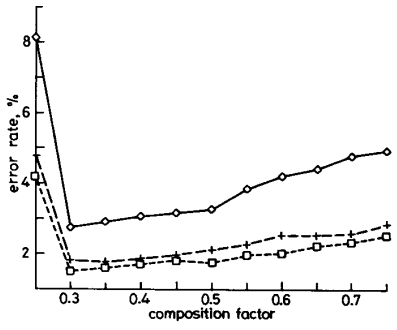

Fig. 2 *Error rate versus composition factor μ (◇ 8C; + 16C; □ 32C)*

error rate around $\mu = 0.3$, 0.35. We will use $\mu = 0.35$ ($\alpha = 0.538$) from now on. Fig. 2 also shows that the probability of quantisation is much more important in recognition than the probability of generation, since for $\mu > 0.3$ the slopes of the plots are smaller than for $\mu < 0.3$.

4 Performance of MVQ modelling

We will show now a comparison of the designed MVQ system with DHMM and SCHMM systems. The DHMM-based system is implemented as described in Reference 7. The SCHMM-based system has been implemented following the description of Reference 2 (using diagonal covariance matrices for each PDF as in EXP1), although carrying out three different types of model training.

SC1: Uses a LBG-trained codebook and the transition and output matrices (A, B) of the discrete model are obtained from a separate Baum–Welch training.

SC2: Uses the joint re-estimation of VQ codebook and the (A, B) matrices described in Reference 2.

SC3: Similar to SC2 but without re-estimation of covariance matrices. According to Huang's work [9], this method can improve the system performance. This result indicates that the acoustic parameters are not correctly modelled by the utilised PDF.

Table 1 shows the error rates obtained for the three types of SCHMMs using 64, 128, 256 and 512 centres in the codebook, and four VQ candidates (the most probable ones) to model each input vector. As suggested in Reference 9, the superiority of SC3 over SC2 indicates that the

Table 1: Error rates for SC1, SC2 and SC3 and several codebook sizes

Number of centres	SC1	SC2	SC3
64/4	3.48	4.11	3.59
128/8	2.76	2.70	2.60
256/16	1.87	2.23	2.10
512/32	1.66	2.29	1.82

estimation of covariance matrices can degrade the performance. It is also observed that SC1 provides similar results to SC3 for 64 and 128 centres, but clearly superior results for 256 and 512 centres. The explanation for this behaviour can be found in two problems that have been previously pointed out: the lack of coherence between probability and distance measures, and the problem of insufficient training when a large number of parameters must be re-estimated (due to the use of one covariance matrix per VQ centre). These results confirm the difficulty, detected by Huang, of modelling the acoustic parameters with multivariate Gaussian PDFs (as in EXP1).

A comparison of the performance achieved by DHMMs, SCHMMs (experiment SC1), and MVQs (as described in the previous section) is shown in Fig. 3.

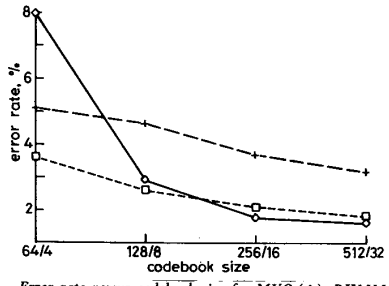

Fig. 3 *Error rate versus codebook size for MVQ (◇), DHMM (+) and SCHMM (□)*

Since the considered vocabulary contains 16 words, the set of 16 N-centre codebooks for MVQ models is equivalent to one $(16 \times N)$-centre codebook for DHMM and SCHMM. Thus, the results are compared when the same total number of centres (4/64, 8/128, 16/256 and 32/512) is used. For the same total number of centres, DHMMs and MVQs require the same amount of computation for recognition, but clearly less than SCHMMs. When using 4/64 centres, the superiority of DHMMs over MVQs is evident, in spite of the loss of signal distortion information during the VQ process. The explanation of this behaviour is straightforward: a four-centre codebook cannot correctly model the acoustic variety of the vocabulary words. However, it is clear that MVQ modelling out-performs DHMM models using eight or more centres, with the same amount of computation in recognition. It must also be pointed out that there is an important computational saving in training, since the computation involved with 16 N-centre codebooks is very much smaller than that of a single $(16 \times N)$-centre codebook using the LBG training algorithm: if the total number of training vectors is K, $K \times (N \times 16)$ distance computations per iteration would be required to train one shared codebook and only $K \times N$ to train 16 MVQ codebooks (assuming that each of them is trained by $K/16$ training vectors). Also, MVQ models can even achieve similar (8/128 centres) or better (from 16/256 centres) results than SCHMMs.

5 SCMVQ-HMM modelling

The results obtained with MVQ modelling suggest the implementation of a new type of models that generalises the MVQ modelling for several quantisation candidates, in the same way as SCHMMs generalise DHMMs. This leads us to the SCMVQ-HMM modelling, for which the output probabilities must be computed as

$$b_o(\boldsymbol{x}) \simeq \sum_{k=1}^{C} P(\boldsymbol{x} \mid v_k, \lambda) b_i(v_k) \tag{9}$$

where C is the number of VQ candidates (in $V(\lambda)$).

This new variant seeks to be a generalisation of MVQ, that is, the SCMVQ modelling must become a MVQ modelling when only one quantisation candidate is utilised. Thus, the PDF utilised must have an analogous form to that in eqn. 7. However, the introduction of the weighting factor α in eqn. 8 must also be taken into account. These considerations lead to the following expression

$$P(\boldsymbol{x} \mid v_k, \lambda) = (2\pi\sigma_\lambda^2)^{-p\alpha/2} \exp\left\{-\frac{\alpha}{2\sigma_\lambda^2}\|\boldsymbol{x} - \boldsymbol{y}_k\|^2\right\} \tag{10}$$

The PDF in eqn. 10 is a non-normalised density due to the introduction of $\alpha = 0.538$. This PDF seeks to keep the advantages previously noted: (1) coherence with the distance measure; and (2) reduction of the number of parameters to re-estimate.

The mechanisms of these new models, for training and recognition, are similar to those of the SCHMM models described in Reference 2, in the same way as MVQs are similar to DHMMs. In a ML estimation of SCMVQs, the VQ parameters can be jointly estimated along with the discrete HMM parameters (A, B and Π matrices). The latter are re-estimated via Baum–Welch, and the VQ parameters (centres and average distortion) of the codebook of model λ by the following re-estimation formulas

$$\hat{\boldsymbol{y}}_k = \frac{\sum_{l=1}^{S(\lambda)} \sum_{t=1}^{T^l} \mathscr{P}_t^l(k) \boldsymbol{x}_t^l}{\sum_{l=1}^{S(\lambda)} \sum_{t=1}^{T^l} \mathscr{P}_t^l(k)} \tag{11a}$$

$$\hat{\sigma}_\lambda^2 = \frac{1}{p} \frac{\sum_{l=1}^{S(\lambda)} \sum_{t=1}^{T^l} \sum_{k=1}^{M} \mathscr{P}_t^l(k) \|\boldsymbol{x}_t^l - \hat{\boldsymbol{y}}_k\|^2}{\sum_{l=1}^{S(\lambda)} \sum_{t=1}^{T^l} \sum_{k=1}^{M} \mathscr{P}_t^l(k)} \tag{11b}$$

where (see Reference 2)

$$\mathscr{P}_t(k) = P(o_t = v_k \mid X, \lambda) \tag{12}$$

and $S(\lambda)$ is the number of training sequences corresponding to model λ. This procedure can be initialised with the LBG codebooks used in the MVQ approach.

It is easy to prove that in the case of using only one quantisation candidate then $\mathscr{P}_t(k) = \delta_{o_t, v_k}$ (δ is the Kronecker delta function), that is, $\mathscr{P}_t(k)$ is different from 0 only in the case that v_k is the symbol at time t. In this case, the dependence of eqn. 11 on the discrete HMM parameters is removed and these equations become the centroid estimation formulas of the LBG algorithm. This is a proof that the estimation previously performed for MVQ models was ML.

The SCMVQ modelling is formally the closest to CHMMs. The only difference is that all the states share the same set $V(\lambda)$ of PDFs. In spite of this similarity to CHMM models, it is easy to understand that the computational complexity of SCMVQ is even smaller than that of SCHMM in both training and recognition, due to the reduction in the number of parameters in the covariance matrices. This computational saving is very significant in the training stage if a joint re-estimation is performed, since the training 16 N-centre codebooks does not

require as much computation as a single (16 × N)-centre codebook (as discussed in Section 4).

5.1 Experimental results

Two different experiments have been carried out with the SCMVQ models described above:

EXP4: Only the A and B matrices are re-estimated, using the same codebooks as for MVQ models.

EXP5: All the model parameters are re-estimated, using formulas 11 (global ML estimation).

These experiments were carried out for 8, 16 and 32 centres per codebook, using from two to eight quantisation candidates (one candidate corresponds to MVQ modelling). The results are shown in Table 2. The effect

Table 2: Error rate values for SCMVQ modelling with 1–8 candidates, for 8 (8C), 16 (16C) and 32 (32C) centres per codebook. Experiments EXP4 and EXP5

Number of candidates	1	2	3	4	5	6	7	8
8C EXP4	2.91	3.02	3.33	3.07	3.17	3.22	3.22	3.22
8C EXP5	—	2.70	2.96	2.81	2.81	2.81	2.81	2.81
16C EXP4	1.77	1.56	1.45	1.51	1.56	1.61	1.61	1.56
16C EXP5	—	1.45	1.51	1.56	1.61	1.66	1.61	1.61
32C EXP4	1.61	1.25	1.40	1.35	1.40	1.40	1.45	1.45
32C EXP5	—	1.25	1.14	1.09	1.09	1.09	1.19	1.14

of including a higher number of candidates depends on the codebook size:

(1) In the case of eight centres (8C), introducing more candidates clearly degrades the system performance when using EXP4. If EXP5 is used, a slight improvement of 0.2% can be achieved with two candidates, although the performance degrades or becomes stable for more candidates.

(2) For 16 centres (16C), EXP4 and EXP5 present similar behaviours, reducing the error rate for two to four candidates.

(3) When using 32 centres (32C), EXP4 reduces again the error rate, although the best results are obtained with EXP5, obtaining an error reduction of 32% (using four to six candidates) (Fig. 4).

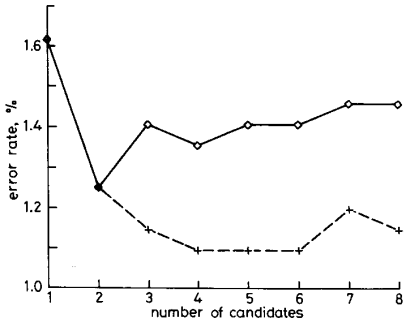

Fig. 4 *Error rate versus number of quantisation candidates for 32-centre codebooks (◇ EXP4; + EXP5)*

Two main conclusions can be extracted:

(*a*) The density of centres must be large enough, in order to improve the recognition results. This means that a given input vector is correctly represented by several VQ candidates only when there is more than one close centre. Thus, it is also very important to select the appropriate number of quantisation candidates depending on the codebook size.

(*b*) It is important to 'teach' the system that other centres, different from the nearest one, can represent a given input vector. This is performed by the joint re-estimation of EXP5. Thus, it possible to avoid the degradation of the eight-centre system, and to obtain meaningful improvements in the case of a high density of centres, as for 32 centres.

Finally, Table 3 shows a comparison of the error rates achieved by DHMM (D), SCHMM (as in SC1), MVQ (M) and SCMVQ (SCM) (as in EXP5). For SCMVQ, two quantisation candidates are used for 8 and 16 centres, and four candidates for 32 centres. In relation to SCHMM, the computational complexity of SCMVQs is always less, due to the use of simplified covariance matrices. This computational reduction is more considerable in the cases of 8 and 16 centres, for which only two candidates are used (four for SCHMMs). It can be observed that MVQs and SCMVQs are always superior for 16/256 and 32/512 centres, the best performance corresponding to SCMVQ (SCMVQ also provides the best result for 8/128). Also, a new experiment (labelled SCN) with SCHMM models has been carried out using joint re-estimation and the same PDFs as in eqn. 10 (σ_1^2 is substituted by the average distortion per feature of the shared codebook). It is interesting to observe that SCN can obtain the same or better results than standard SCHMM (using more than 64 centres), although it is only superior to SCMVQs for 8/128 centres. This new variation uses the same amount of computation as SCMVQ in recognition, but, again, more in training. This result ratifies the suitability of the PDF given by eqn. 10 for speech recognition.

Table 3: Error rate for DHMM(D), SCHMM(SC1), MVQ(M), SCMVQ(SCM) and experiment SCN

Number of centres	D	SC1	M	SCM	SCN
64/4	5.10	3.48	7.96	—	4.01
128/8	4.63	2.76	2.91	2.70	2.34
256/16	3.69	1.87	1.77	1.45	1.87
512/32	3.17	1.66	1.61	1.09	1.40

6 Summary and future work

We have introduced in this paper a new type of HMM, called SCMVQ-HMM. It is a generalisation of the MVQ modelling recently introduced to enhance discrete HMMs with the spectral information lost in the VQ process. We looked first for an appropriate form for the PDFs in a MVQ system. The chosen PDF has three main features:

(*a*) it reduces the required number of model parameters, lightening the insufficient training problem

(*b*) it is coherent with the utilised distance measure

(*c*) it is weighted for an optimal composition with discrete HMM probabilities.

Comparison of the MVQ system with standard DHMMs and SCHMMs has shown the potential of this approach. The SCMVQ modelling generalises MVQ using several quantisation candidates. In the same way as for

SCHMM models, the ML estimation of SCMVQs allows the joint re-estimation of the VQ and the discrete HMM parameters. The results with SCMVQ models show that it is very important to have a codebook size large enough to correctly model an input vector with several quantisation candidates, and that an appropriate selection of the number of candidates must be made. It is also important to train the system to use several candidates by means of the joint re-estimation. SCMVQ modelling can obtain an error-rate reduction of 32% with respect to MVQ. The SCMVQs outperforms all the types of HMMs previously tested (DHMMs, SCHMMs and MVQs) using less computation than SCHMMs in both training and recognition, due to the parameter reduction in the PDFs utilised. There is also an additional decrease of computation (by a factor equal to the number of recognition units) in the training stage due to the management of smaller codebooks (assuming that the global number of centres is kept the same).

We believe that the application of the techniques introduced in this paper to a CSR task can be carried out straightforwardly, associating different codebooks to different subword units. For example, assuming phoneme-like units and that each of our 16 words contains different phonemes, we estimate that an appropriate phoneme codebook size could vary between four and eight (this is roughly equivalent to the use of word codebooks with 16–32 centres), just enough to model the transition/stable-part/transition sequence corresponding to a certain phoneme (several transitions could be taken into account with different codebook centres) and to accomplish the 'high density' requirement of SCMVQ. In fact, our research group is currently working on the application of basic MVQ models to CSR, and the preliminary results are encouraging.

7 References

1 RABINER, L.R.: 'A tutorial on hidden Markov models and selected applications in speech recognition', *Proc. IEEE*, 1989, **77**, pp. 257–285

2 HUANG, X., and JACK, M.: 'Unified techniques for vector quantisation and hidden Markov modeling using semi-continuous models'. Proceedings of ICASSP-89, Glasgow, Scotland, May 1989, pp. 639–642

3 SEGURA, J., RUBIO, A., PEINADO, A., GARCÍA, P., and ROMÁN, R.: 'Multiple VQ hidden Markov modelling for speech recognition', *Speech Commun.*, 1994, **14**, pp. 163–170

4 PEINADO, A., SEGURA, J., RUBIO, A., and SÁNCHEZ, V.: 'A MMI codebook design for MVQHMM speech recognition', *in* 'New advances and trends in speech recognition and coding' (NATO ASI 93, in press)

5 LINDE, Y., BUZO, A., and GRAY, R.: 'An algorithm for vector quantizer design', *IEEE Trans. Commun.*, 1980, **28**, pp. 84–95

6 LEVINSON, S., Rabiner, L., and SONDHI, M.: 'An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition', *Bell System Technical J.*, 1983, **62**, pp. 1035–1074

7 PEINADO, A., LÓPEZ, J., SÁNCHEZ, V., SEGURA, J., and RUBIO, A.: 'Improvements in HMM-based isolated word recognition system', *IEE Proc. I*, 1991, **138**, pp. 201–206

8 BURTON, D., SHORE, J., and BUCK, J.: 'Isolated-word speech recognition using multisection vector quantization codebooks', *IEEE Trans. ASSP*, 1985, **33**, pp. 837–849

9 HUANG, X., LEE, K., and HON, H.: 'On semi-continuous hidden Markov modelling'. Proceedings of ICASSP-90, 1990, pp. 689–692