COMBINING VTS MODEL COMPENSATION AND SUPPORT VECTOR MACHINES

M.J.F. Gales and F. Flego

Cambridge University Engineering Department
Trumpington St., Cambridge CB2 1PZ, U.K.
{mjfg,ff257}@eng.cam.ac.uk

ABSTRACT

It is difficult to adapt discriminative classifiers, particularly kernel based ones such as support vector machines (SVMs), to handle mismatches between the training and test data. In previous work adaptation was performed by modifying the kernel used with the SVM, rather changing the SVM parameters themselves. However an idealised form of compensation, single pass retraining, was used to alter the generative models associated with the generative kernel. In this paper vector Taylor series model compensation is used. This scheme is more efficient and allows a noise model to be estimated. The performance of the new scheme is evaluated on two continuous digit tasks. On both tasks SVM-rescoring outperformed the baseline VTS compensated models.

Index Terms: speech recognition, noise robustness, support vector machines, vector Taylor series compensation.

1. INTRODUCTION

Speech recognition is normally based on generative models, in the form of Hidden Markov Models (HMMs), and class priors, the language model. These are then combined using Bayes' decision rule. An alternative approach is to use discriminative models, or discriminative functions such as Support Vector Machines (SVMs) [1]. One of the problems with using these discriminative models and functions is that it is normally hard to adapt them to changing speakers or acoustic environments. This is particularly true of kernel based approaches, such as SVMs, where individual training examples are used to determine the decision boundaries. One approach to handling SVM-based adaptation is described in [2]. This involves using the support vectors from the original, unadapted, model in combination with the adaptation data.

An obvious application area where there are large mismatches between the training and test sets is speech recognition in noise. Handling changing acoustic conditions has been an active area of research for many years. Model-based compensation schemes [3, 4, 5] are a powerful approach to handling mismatches between training and test conditions. Well implemented model-based compensation schemes tend to out-perform feature-based compensation schemes as it is possible to more accurately model situations where speech is, for example, masked by the noise.

In previous work [6], SVMs were adapted to differing noise environments by using noise-specific generative kernels. Generative kernels use feature spaces defined by generative models, in this case HMMs. By adapting these generative models to the changing noise conditions allows noise-specific kernels to be produced. The model-based compensation used was single-pass retraining (SPR) [3]. This is an idealised form of model-based compensation which is impractical for most applications as it requires the background noise to be known and all the training data available. In this work the same approach for adapting the SVMs to the noise conditions is used, but the compensation is performed using Vector Taylor Series (VTS) model compensation with maximum likelihood noise estimation [7]. This is a more practical scheme than SPR. The combination of SVM rescoring with VTS is evaluated on the AURORA task as in the previous work. In addition it is evaluated on data recorded in-car by Toshiba Research Europe Ltd (TREL). This new data allowed the evaluation on more realistic data, as well as how the scheme may be used with sub-word units.

2. MODEL-BASED NOISE COMPENSATION

The first stage in producing a noise compensation scheme is to define the impact of the acoustic environment and channel on the clean speech data, the *mismatch function*. In the mel-cepstral domain used in this work the following approximation between the static clean speech, noise and noise corrupted speech observations is used ($\log(.)$ and $\exp(.)$ indicate element-wise logarithm or exponential functions)

$$\begin{aligned} \boldsymbol{y}_t^{\mathsf{s}} &= \boldsymbol{x}_t^{\mathsf{s}} + \boldsymbol{h} + \mathbf{C}\log\left(\mathbf{1} + \exp(\mathbf{C}^{-1}(\boldsymbol{n}_t^{\mathsf{s}} - \boldsymbol{x}_t^{\mathsf{s}} - \boldsymbol{h}))\right) \\ &= \boldsymbol{x}_t^{\mathsf{s}} + \boldsymbol{h} + \boldsymbol{f}(\boldsymbol{n}_t^{\mathsf{s}} - \boldsymbol{x}_t^{\mathsf{s}} - \boldsymbol{h}) \end{aligned} \quad (1)$$

where $\mathbf{C}$ is the DCT matrix[1]. For a given set of noise conditions, the observed (static) speech vector $\boldsymbol{y}_t^{\mathsf{s}}$ is a highly non-linear function of the underlying clean (static) speech signal $\boldsymbol{x}_t^{\mathsf{s}}$, noise $\boldsymbol{n}_t^{\mathsf{s}}$ and convolutional noise $\boldsymbol{h}$. Noise compensation schemes are further complicated by the addition of dynamic parameters. The observation vector $\boldsymbol{y}$ is often formed of the static parameters appended by the delta and delta-delta parameters. Thus $\boldsymbol{y}_t^{\mathsf{T}} = \left[\begin{array}{ccc} \boldsymbol{y}_t^{\mathsf{sT}} & \Delta\boldsymbol{y}_t^{\mathsf{sT}} & \Delta^2\boldsymbol{y}_t^{\mathsf{sT}} \end{array}\right]$. Mismatch functions for all the parameters can be obtained [7].

The aim of model-based compensation schemes is to obtain the parameters of the noise-corrupted speech model from the clean speech and noise models. Most model-based compensation methods assume that if the speech and noise models are Gaussian then the combined noisy model will also be Gaussian. Thus to compute the expected value of the observation for each clean speech component (assuming a single noise component) the following must be computed

$$\boldsymbol{\mu}_{\mathsf{y}}^{(m)} = \mathcal{E}\left\{\boldsymbol{y}\right\}; \quad \boldsymbol{\Sigma}_{\mathsf{y}}^{(m)} = \mathrm{diag}\left(\mathcal{E}\left\{\boldsymbol{y}\boldsymbol{y}^{\mathsf{T}}\right\} - \boldsymbol{\mu}_{\mathsf{y}}^{(m)}\boldsymbol{\mu}_{\mathsf{y}}^{(m)\mathsf{T}}\right) \quad (2)$$

where the expectation is over the clean speech "observations" from component m and noise "observations" combined using equation 1.

This work was partly funded by Toshiba Research Europe Ltd.

[1] For a discussion of variations on this mismatch function see [8].

There is no simple closed-form solution to these equations so various approximations have been proposed. These include Parallel Model Combination [3] and Vector Taylor Series [4]. An additional problem that must be solved is that noise models are not normally available. Thus these must be estimated from the observed data.

Vector Taylor series model-based compensation is a popular approach for model-based compensation [4, 5, 7, 9]. There are a number of possible forms that have been examined. In this work the first-order VTS scheme described in [7] is used. A brief summary of the scheme is given here. The static mean, $\boldsymbol{\mu}_{\mathsf{y}}^{\mathsf{s}}$, and covariance matrix, $\boldsymbol{\Sigma}_{\mathsf{y}}^{\mathsf{s}}$, of the corrupted speech distribution are given by [9][2]

$$\boldsymbol{\mu}_{\mathsf{y}}^{\mathsf{s}} = \boldsymbol{\mu}_{\mathsf{x}}^{\mathsf{s}} + \boldsymbol{\mu}_{\mathsf{h}} + \boldsymbol{f}(\boldsymbol{\mu}_{\mathsf{n}}^{\mathsf{s}} - \boldsymbol{\mu}_{\mathsf{x}}^{\mathsf{s}} - \boldsymbol{\mu}_{\mathsf{h}}) \quad (3)$$

$$\boldsymbol{\Sigma}_{\mathsf{y}}^{\mathsf{s}} = \text{diag}\left(\mathbf{A}\boldsymbol{\Sigma}_{\mathsf{x}}^{\mathsf{s}}\mathbf{A}^{\mathsf{T}} + (\mathbf{I}-\mathbf{A})\boldsymbol{\Sigma}_{\mathsf{n}}^{\mathsf{s}}(\mathbf{I}-\mathbf{A})^{\mathsf{T}}\right) \quad (4)$$

where matrix $\mathbf{A}$ above is the partial derivative, $\partial \boldsymbol{y}^{\mathsf{s}}/\partial \boldsymbol{x}^{\mathsf{s}}$, evaluated at $\overline{\boldsymbol{\mu}}^{\mathsf{s}} = \boldsymbol{\mu}_{\mathsf{n}}^{\mathsf{s}} - \boldsymbol{\mu}_{\mathsf{x}}^{\mathsf{s}} - \boldsymbol{\mu}_{\mathsf{h}}$. This may be expressed as

$$\mathbf{A} = \partial \boldsymbol{y}^{\mathsf{s}}/\partial \boldsymbol{x}^{\mathsf{s}} = \mathbf{C}\mathbf{F}\mathbf{C}^{\text{-}1} \quad (5)$$

where $\mathbf{F}$ is a diagonal matrix with elements given by $\mathbf{1}/(\mathbf{1} + \exp(2\mathbf{C}^{\text{-}1}(\overline{\boldsymbol{\mu}}^{\mathsf{s}}))$. Similar expressions can be found for the dynamic parameter compensation using the *continuous time approximation*.

The compensation schemes described above have assumed that the noise model parameters, $\boldsymbol{\mu}_{\mathsf{n}}$, $\boldsymbol{\Sigma}_{\mathsf{n}}$ and $\boldsymbol{\mu}_{\mathsf{h}}$, are known. In practice these are seldom known in advance so must be estimated from the test data. In this work the noise estimation is based on the Maximum Likelihood (ML) noise estimation scheme described in [7]. In addition, the Hessian approach for the noise variance in [5] was implemented. This has no effect on recognition performance, but improves the estimation speed as there are fewer back-offs to ensure that the auxiliary function increases.

3. SVMs AND GENERATIVE KERNELS

Support Vector Machines (SVMs) [1] are an approximate implementation of structural risk minimisation. They have been found to yield good performance on a wide range of tasks. The theory behind SVMs has been extensively described in many papers and is not discussed here. This section concentrates on how SVMs can be applied to tasks where there is sequence data, for example speech recognition.

One of the issues with applying SVMs to sequence data, such as speech, is that the SVM is inherently static in nature; "observations" (or sequences) are all required to be of the same dimension. A range of *dynamic kernels* have been proposed that handle this problem. Of particular interest in this work are those kernels that are based on generative models [10, 11]. In these approaches a generative model is used to determine the feature-space for the kernel. An example first-order feature-space for a generative kernel with observation sequence $\mathbf{Y}$ may be written as

$$\boldsymbol{\phi}(\mathbf{Y};\boldsymbol{\lambda}) = \frac{1}{T}\begin{bmatrix} \log\left(p(\mathbf{Y};\boldsymbol{\lambda}^{(\omega_1)})\right) - \log\left(p(\mathbf{Y};\boldsymbol{\lambda}^{(\omega_2)})\right) \\ \boldsymbol{\nabla}_{\lambda^{(\omega_1)}} \log p(\mathbf{Y};\boldsymbol{\lambda}^{(\omega_1)}) \\ \boldsymbol{\nabla}_{\lambda^{(\omega_2)}} \log p(\mathbf{Y};\boldsymbol{\lambda}^{(\omega_2)}) \end{bmatrix} \quad (6)$$

where $p(\mathbf{Y};\boldsymbol{\lambda}^{(\omega_1)})$ and $p(\mathbf{Y};\boldsymbol{\lambda}^{(\omega_2)})$ are the likelihood of the data using generative models associated with classes ω_1 and ω_2 respectively. HMMs are used as the generative model in this paper. Considering only the derivative with respect to the means, the feature-space will have the form

$$\frac{\partial}{\partial \boldsymbol{\mu}_m^{(\omega_1)}} \log p(\mathbf{Y};\boldsymbol{\lambda}^{(\omega_1)}) = \sum_{t=1}^{T} \gamma_m(t)\boldsymbol{\Sigma}_m^{(\omega_1)\text{-}1}\left(\boldsymbol{y}_t - \boldsymbol{\mu}_m^{(\omega_1)}\right) \quad (7)$$

where $\gamma_m(t)$ is the posterior probability that component m generated the observation at time t given the complete observation sequence $\mathbf{Y}$. Only the derivatives with respect to the means are used in this work, though it is possible to use other, and higher-order, derivatives. As SVM training is a distance based learning scheme it is necessary to define an appropriate metric for the distance between two points. In this work a maximally non-committal metric is used.

$$K(\mathbf{Y}_i,\mathbf{Y}_j;\boldsymbol{\lambda}) = \boldsymbol{\phi}(\mathbf{Y}_i;\boldsymbol{\lambda})^{\mathsf{T}}\mathbf{G}^{\text{-}1}\boldsymbol{\phi}(\mathbf{Y}_j;\boldsymbol{\lambda}) \quad (8)$$

where $\mathbf{Y}_i$ and $\mathbf{Y}_j$ are two observation sequences and $\mathbf{G}$ is related to the Fisher Information matrix (the log-likelihood ratio is also included). In common with other work in this area [10, 11], $\mathbf{G}$ is approximated by the diagonalised empirical covariance matrix of the training data.

Classification using this form of generative kernel with observation sequence $\mathbf{Y}$ and training data $\mathbf{Y}_1,\ldots,\mathbf{Y}_n$ is then based on the SVM score $\mathcal{S}(\mathbf{Y})$

$$\mathcal{S}(\mathbf{Y}) = \sum_{i=1}^{n} \alpha_i^{\text{svm}} z_i K(\mathbf{Y}_i,\mathbf{Y};\boldsymbol{\lambda}) + b \quad (9)$$

$$\hat{\omega} = \begin{cases} \omega_1, & \mathcal{S}(\mathbf{Y}) \geq 0 \\ \omega_2, & \mathcal{S}(\mathbf{Y}) < 0 \end{cases} \quad (10)$$

where α_i^{svm} is the Lagrange multiplier for observation sequence $\mathbf{Y}_i$ obtained from the SVM maximum margin training, b is the bias and $z_i \in \{1,-1\}$ indicates whether the sequence was a positive (ω_1) or negative (ω_2) example.

4. SVMs FOR NOISE ROBUSTNESS

The previous two sections have described VTS model compensation and support vector machines with generative kernels. This section describes how these schemes can be combined together to allow noise-specific generative kernels to be used with a noise-independent SVM for speech recognition. The process for training and evaluating the SVMs is similar to the one described in [6]. The main difference is that one-vs-one majority is used to handle the multi-class problem.

The procedure for training the noise-independent SVMs is:

1. For each training condition perform model compensation
2. Align all the training utterances $\mathbf{Y}_1,\ldots,\mathbf{Y}_n$ using reference, $\mathbf{r} = r_1,\ldots,r_K$ to give the word-segmented data sequence $\tilde{\mathbf{Y}}_1,\ldots,\tilde{\mathbf{Y}}_K$
3. For each confusable pair (ω_l,ω_j) set $\boldsymbol{\lambda} = \{\boldsymbol{\lambda}^{(\omega_l)},\boldsymbol{\lambda}^{(\omega_j)}\}$
   (a) obtain $\boldsymbol{\phi}(\tilde{\mathbf{Y}}_i;\boldsymbol{\lambda})$ for all training examples of ω_l using the appropriate noise compensated $\boldsymbol{\lambda}$
   (b) obtain $\boldsymbol{\phi}(\tilde{\mathbf{Y}}_i;\boldsymbol{\lambda})$ for all training examples of ω_j using the appropriate noise compensated $\boldsymbol{\lambda}$
   (c) train a noise-independent SVM for pair (ω_l,ω_j) using all positive (a) and negative (b) examples.

In this work only the log-likelihood ratio and derivatives with respect to the means are used. There is an issue with directly using equation 7. Model-based compensation schemes normally modify the variances of the acoustic models. To keep the dynamic ranges

[2]The dependence on the noise corrupted speech mean and clean speech mean on the component have been dropped for clarity.

of each set of features consistent standard-deviation normalisation, rather than the variance normalisation in equation 7, is used. Note this is not normally a problem as the same covariance matrices are used for all sequences and the dynamic-range effects handled by the metric $\mathbf{G}$.

During recognition the following procedure is used:

1. Compensate the acoustic models for the test condition
2. Recognise the test utterance $\mathbf{Y}$ to obtain 1-best hypothesis, $\mathbf{h} = h_1, \ldots, h_K$ and align to give the word-segmented data sequence $\tilde{\mathbf{Y}}_1, \ldots, \tilde{\mathbf{Y}}_K$
3. For each segment $\tilde{\mathbf{Y}}_i$:

a) for each word pair $\{\omega_l, \omega_j\}$ set $\boldsymbol{\lambda} = \{\boldsymbol{\lambda}^{(\omega_l)}, \boldsymbol{\lambda}^{(\omega_j)}\}$

$$\hat{\omega} = \begin{cases} \omega_l, & \text{if } \mathcal{S}(\tilde{\mathbf{Y}}_i) + \frac{\epsilon}{\sqrt{g_{11}}} \log\left(\frac{p(\tilde{\mathbf{Y}}_i;\boldsymbol{\lambda}^{(\omega_l)})}{p(\tilde{\mathbf{Y}}_i,\boldsymbol{\lambda}^{(\omega_j)})}\right) \geq 0 \\ \omega_j; & \text{otherwise} \end{cases} \quad (11)$$

count $[\hat{\omega}] =$ count $[\hat{\omega}] + 1$

b) classification, $\hat{h}_i$, is given by:

1) if no ties in voting: $\hat{h}_i = \text{argmax}_\omega\{\text{count}[\omega]\}$

2) if only two words (w_l, w_j) tie then $\hat{h}_i$ determined using the result from that pair in equation 11

3) if more than two words tie $\hat{h}_i = h_i$

ϵ is used to scale the contribution of the log-likelihood ratio to the SVM score. The log-likelihood ratio is the most discriminatory of the dimensions of the score-space. However using a maximally non-committal metric, $\mathbf{G}$, all dimensions are treated equally. Thus ϵ is used to reflect the usefulness of the log-likelihood ratio. As $\epsilon \to \infty$ the performance of the system will tend to the HMM performance. In these experiments ϵ was normalised using the first element of the metric $\mathbf{G}$ (associated with the log-likelihood ratio). Though in preliminary experiments this made little different, it is felt that it makes the choice of ϵ less sensitive to the task.

5. RESULTS

Two continuous digit recognition tasks were used to evaluate the combination of VTS with the proposed SVM rescoring scheme. The first AURORA 2 is a database where noise has been artificially added to clean data. The second task used in-car data recorded by Toshiba Research Europe Ltd. For both tasks HTK frontend was used to derive 39 dimensional feature vectors consisting of 12 MFCCs appended with the zeroth cepstrum, delta and delta-delta coefficients. The VTS approach adopted was similar to the procedure in [5]. An initial estimate of the background additive noise for each utterance was obtained using the first and last 20 frames of the utterance. This was then used as the noise model for VTS compensation and each utterance recognised. This hypothesis was used to estimate a per-utterance noise model in an ML-fashion. This process was then optionally repeated. The final recognition output used this ML-estimated noise model for VTS compensation. For all SVM rescoring experiments the SVMs were built using the top 1500 dimensions of $\boldsymbol{\phi}(\tilde{\mathbf{Y}}_i; \boldsymbol{\lambda})$ ranked using the Fisher ratio and ϵ was set to 2.

5.1. AURORA 2

AURORA 2 is a small vocabulary digit string recognition task [12]. As the vocabulary size (excluding silence) is only eleven (one to nine, plus zero and oh) the number of word pairs is small (66 including silence) making it suitable for the proposed scheme. The utterances in this task are one to seven digits long based on the TIDIGITS database with noise artificially added. The clean training data was used to train the acoustic models. This comprises 8440 utterances from 55 male and 55 female speakers. The acoustic models are 16 emitting states whole word digit models, with 3 mixtures per state and silence and inter-word pause models. All three test sets, A,B and C, were used for evaluating the schemes. For sets A and B, there were a total of 8 noise conditions (4 in each) at 5 different SNRs, 0dB to 20dB. For test set C there were two additional noise conditions at the same range of SNRs. In addition to background additive noise convolutional distortion was added to test set C. Test set A was used as the development set for tuning parameters.

For the SVM rescoring experiments, the SVMs were trained on a subset of the multi-style training data available for the noise conditions and SNRs in test set A. For each of the noise/SNR conditions there are 422 sentences (a subset of all the training data). For the SVMs training only three of the four available noise conditions (N2-N4) and three of the five SNRs 10dB, 15dB and 20dB were used. This allows the generalisation of the SVM to unseen noise conditions to be evaluated on test set A as well as the test sets B and C.

SNR (dB)	Set A		Set B		Set C	
	VTS	SVM	VTS	SVM	VTS	SVM
20	1.69	1.35	1.46	1.22	1.57	1.33
15	2.36	1.82	2.37	1.77	2.47	2.00
10	4.39	3.23	4.12	3.16	4.49	3.52
05	11.20	8.22	10.05	7.68	10.69	8.70
00	29.55	23.00	27.54	22.93	28.41	25.01
Avg	9.84	7.52	9.11	7.35	9.53	8.11

Table 1. VTS (1 iteration) and SVM rescoring performance WER (%) tests Sets A, B, C ($\epsilon = 2$), SVMs trained on test set A N2-N4 10-20dB SNR.

Table 1 summarises the results for VTS compensation and SVM rescoring for all three available test sets[3]. For all noise conditions large reductions in WER were obtained using SVM rescoring compared to the baseline VTS compensation. Though the relative gains for test sets B and C were slightly less than that for test set A, it still indicate that a good level of noise-independent classification can be obtained using these noise-specific generative kernels.

5.2. Toshiba In-Car Data

The scheme was also evaluated on a task with real recorded noise: the Toshiba in-car database. This is a corpus collected by Toshiba Research Europe Limited's Cambridge Research Laboratory. It is a small/medium sized task with noisy speech collected in the office and in vehicles driving at various conditions. This work uses three of the test sets containing digit sequences (phone numbers) recorded in a car with a microphone mounted on the rear-view mirror. The ENON set, which consists of 835 utterances, is recorded with the engine idle, and has a 35 dB average SNR. The CITY set, which consists of 862 utterances, is recorded driving in cities, and has a 25 dB average SNR. The HWY set, which consists of 887 utterances, is recorded on the highway, and has a 18 dB average SNR. Noise compensation was applied to a speech recogniser trained on clean data from the Wall Street Journal (WSJ) corpus. The total number of states was about 650 with 12 Gaussian components per state. This

[3]For a more detailed discussion of the set-up and improved performance using the ETSI frontend and optimised mismatch function see [8].

system is more compact than the usual form of system built on the WSJ data, but is felt to be more realistic for an embedded application whilst maintaining the flexibility to be applicable to a wide-range of tasks. For the initial decoding the acoustic models were decision tree clustered state, cross-word triphones, with three emitting states per HMM, twelve components per GMM and diagonal covariance matrices. In addition to the clean system, a multi-style trained system was built. The training data for this system was generated by adding car-noise onto the WSJ data. For additional information about the noise sources and SNRs see [13].

System	VTS iter	Condition WER (%)		
		ENON	CITY	HWY
Clean	—	3.85	31.81	66.18
VTS	1	1.24	3.09	3.78
+SVM		1.25	2.60	3.17
VTS	2	1.37	2.65	3.15
+SVM		1.31	2.14	2.48
MST	—	2.71	6.82	27.50

Table 2. Clean, VTS, SVM rescoring ($\epsilon = 2$) and Multi-style trained (MST) system performance on the Toshiba in-car task.

Table 2 shows the performance of VTS on the Toshiba in-car data. As expected there are large gains over the unadapted Clean system performance and even the system trained in a multi-style fashion. It is not possible to do SVM rescoring with the cross-word triphone models as whole word SVM parameters are required. To handle this a word-internal system was built using the same data. These sub-word models could then be combined to form context-independent whole-word models. This allows models with the same complexity as the baseline system to be constructed. Note some words, such as "OH" only occur in the WSJ training data 89 times (including phrases such as "OH NO" rather than number sequences). Applying SVM rescoring using generative kernels based on these word-internal models gave gains on the two lower SNR condition sets CITY and HWY and no difference in performance for the ENON condition. VTS iteration 2 results were obtained after re-estimation of the noise parameters based on recognition hypothesis from VTS iteration 1. Better noise estimates could have been obtained if the recognition hypothesis from SVM rescoring of VTS iteration 1 was used. As different acoustic models are used for the SVM kernel, there is a small cross system effect. Thus for the HWY condition setting $\epsilon = \infty$ gave a WER of 3.01% for VTS iteration 2. Thus SVM rescoring gave gains of over 10% relative reduction in WER over VTS taking into account cross-system effects.

It is also interesting to see what happens when only a subset of the SVM pairs are used. This is more similar to only rescoring highly confusable pairs as done, for example, in [6]. The subset of SVMs was chosen according to the confusion matrix obtained after VTS iteration 2 of the previous experiment. Using only the 5 most confusable pairs (in order of the number of errors) for rescoring achieved a WER of 2.80% on the HWY condition (VTS iteration 2). Thus the approach is useful where the number of classes makes the current simple one-vs-one majority voting scheme impractical.

6. CONCLUSIONS

This paper has described how vector Taylor series model compensation can be used in combination with SVM rescoring to improve noise robustness. VTS is used to compensate the acoustic models, in this case HMMs, which are then used to define feature-spaces for noise specific kernels. These so-called generative kernels allow the variable length speech data to be mapped to a fixed dimensional vector. This scheme was then evaluated on two digit string recognition tasks. To handle the multi-class problem a rescoring process was used. Hypothesised word boundaries were identified and then majority voting applied given those boundaries. This allows continuous digits, rather than just isolated digits, to be rescored.The AURORA 2 task is a standard task with noise artificially added to clean digit strings. The second task was recorded in-car, thus being more realistic. On both task SVM-rescoring out-performed the baseline VTS compensated acoustic models. In addition rescoring only a highly confusable subset still yielded gains. The baseline models used for these experiments were trained using ML. Using discriminative criteria to train the HMM system may yield gains for the baseline VTS configuration. However, it is expected that there will then be additional performance gains for SVM rescoring.

7. REFERENCES

[1] VN Vapnik, *Statistical learning theory*, John Wiley & Sons, 1998.

[2] X Li and J Bilmes, "Regularized adaptation of discriminative classifiers," in *Proc. ICASSP*, Toulouse, France, 2006.

[3] MJF Gales, *Model-based Techniques for Noise Robust Speech Recognition*, Ph.D. thesis, Cambridge University, 1995.

[4] PJ Moreno, *Speech Recognition in Noisy Environments*, Ph.D. thesis, Carnegie Mellon University, 1996.

[5] J Li, L Deng, Y Gong, and A Acero, "HMM adaptation with joint compensation of additive and convolutive distortions via vector Taylor series," in *ASRU 2007*, Kyoto, Japan, 2007.

[6] MJF Gales and C Longworth, "Discriminative classifiers with generative kernels for noise robust ASR," in *Proc. InterSpeech*, Brisbane, Australia, 2008.

[7] H Liao and MJF Gales, "Joint uncertainty decoding for robust large vocabulary speech recognition," Tech. Rep. CUED/F-INFENG/TR552, Cambridge University, November 2006, Available from: http://mi.eng.cam.ac.uk/∼mjfg.

[8] MJF Gales and F Flego, "Discriminative classifiers with generative kernels for noise robust speech recognition," Tech. Rep. CUED/F-INFENG/TR605, Cambridge University, August 2008, Available from: http://mi.eng.cam.ac.uk/∼mjfg.

[9] A Acero, L Deng, T Kristjansson, and J Zhang, "HMM Adaptation using Vector Taylor Series for Noisy Speech Recognition," in *Proc. ICSLP*, Beijing, China, 2000.

[10] ND Smith and MJF Gales, "Speech recognition using SVMs," in *Advances in Neural Information Processing Systems*, 2001.

[11] M Layton, *Augmented Statistical Models for Classifying Sequence Data*, Ph.D. thesis, Cambridge University, 2006.

[12] H-G Hirsch and D Pearce, "The AURORA experimental framework for the evaluation of speech recognition systems under noisy conditions," in *ASR-2000*, 2000.

[13] H Liao, *Uncertainty Decoding For Noise Robust Speech Recognition*, Ph.D. thesis, Cambridge University, 2007.