EXPERIMENTS WITH LINEAR AND NONLINEAR FEATURE TRANSFORMATIONS IN HMM BASED PHONE RECOGNITION

Panu Somervuo

International Computer Science Institute
Berkeley, California, USA
panus@icsi.berkeley.edu

Neural Networks Research Centre
Helsinki University of Technology, Finland
panu.somervuo@hut.fi

ABSTRACT

Feature extraction is the key element when aiming at robust speech recognition. In this work both linear and nonlinear data-driven feature transformations were applied to the logarithmic mel-spectral context feature vectors in the TIMIT phone recognition task. Transformations were based on Principal Component Analysis (PCA), Independent Component Analysis (ICA), Linear Discriminant Analysis (LDA) and multilayer perceptron network based Nonlinear Discriminant Analysis (NLDA). All four methods outperformed the baseline system which consisted of the standard feature representation based on MFCCs with the first-order deltas, using a mixture-of-Gaussians HMM recognizer. Further improvement was gained by forming the feature vector as a concatenation of the outputs of all four feature transformations.

1. INTRODUCTION

Feature transformations can be divided into two main categories: unsupervised and discriminative. Inside these classes the transformation can be linear or nonlinear. Linear transformations can be implemented by matrix multiplications and nonlinear transforms by using e.g. MLP networks.

In this work four feature transformations, three linear and one nonlinear, were experimented in the TIMIT phone recognition task. In each transformation, the input feature vector was a five-frame window of successive logarithmic mel-spectrum vectors. Two linear transforms based on Principal Component Analysis (PCA) and Independent Component Analysis (ICA) were unsupervised in nature, i.e., the class information of the training data was not used when forming them. In two other transformations based on Linear Discriminant Analysis (LDA) and its nonlinear extension (NLDA) implemented by an MLP network, the class information was utilized. The baseline system consisted of standard MFC features with the first-order deltas.

This work was supported by the Academy of Finland, project no. 44886 “New information processing principles” (Finnish Centre of Excellence Programme 2000-2005).

2. FEATURE TRANSFORMATIONS

The basic ideas behind the experimented data-driven feature transformations are described in this section. For text book references, see e.g. [1] and [2]. The dimensionality of the feature vector before the transformation is denoted by D and after the transformation by D'.

2.1. Principal Component Analysis

Principal Component Analysis is a method to represent the data in the low-dimensional subspace. The corresponding projection matrix is called Karhunen-Loeve transform (KLT). When the original feature vectors are projected into a lower-dimensional linear subspace using KLT, the reconstruction error is the smallest possible among linear transformations. The reconstruction error is measured as the mean-square error between the data vectors in the original feature space and in the projection space. The rows of the $D' \times D$ KLT transformation matrix consist of the D' eigenvectors corresponding to the D' largest eigenvalues of the covariance matrix of the training data. These eigenvectors are the principal axes of the data set. KLT decorrelates the feature vectors, which enables modeling the data with diagonal Gaussians.

2.2. Independent Component Analysis

The idea behind using the Independent Component Analysis in the feature extraction is to reduce the redundancy of the original feature vector components. While PCA removes the second order dependencies of the features vector components, ICA removes also higher order dependencies (minimizes the mutual information between the feature vector components).

The data model of the linear ICA is $\mathbf{x} = \mathbf{A}\mathbf{s}$, where $\mathbf{x}$ is the original feature vector, $\mathbf{s}$ is the vector of the underlying (independent) sources, and $\mathbf{A}$ is a mixing matrix. Only $\mathbf{x}$ is observed, and the goal is to estimate both $\mathbf{A}$ and $\mathbf{s}$ trying to find the sources $\mathbf{s}$ which are statistically independent.

The column vectors of the mixing matrix correspond to the building blocks of the data in the generative model. When the mixing matrix $\mathbf{A}$ has been estimated from the training data, the transformation matrix for obtaining a new feature representation is its inverse, $\mathbf{W} = \mathbf{A}^{-1}$. When the data vector $\mathbf{x}$ is projected to the row vectors of $\mathbf{W}$, the components of the new feature vector represent the activations of the sources $\mathbf{s}$. ICA representation is usually sparse, i.e., only few sources are active at the same time.

There are only few previous work using ICA in the feature transformation related to speech recognition. In [3], ICA was compared to the standard MFCC, but no comparison between PCA and ICA was then made. Also, in that work ICA was applied only to the single-frame feature vectors. In the current work ICA is applied to the multi-frame context windows, see Fig. 1.

2.3. Linear Discriminant Analysis

LDA attempts to separate classes using linear hyperplanes. Such basis vectors are sought which try to maximize the linear class separation. Class separability is measured by the within-class variance and between-class variance. The former is tried to be minimized while the latter is tried to be maximized. Two covariance matrices are computed, the within-class covariance matrix $\mathbf{S}_w$ and the between-class covariance matrix $\mathbf{S}_b$. D' linear discriminants are obtained by taking the eigenvectors corresponding to the D' largest eigenvalues of the matrix $\mathbf{S}_w^{-1}\mathbf{S}_b$ (for c classes there are at most $c-1$ linearly independent eigenvectors). In the feature transformation, the original feature vectors are projected to these eigenvectors. For previous work on LDA applied to the context feature vectors, see e.g. [4].

2.4. Nonlinear Discriminant Analysis

NLDA is a nonlinear extension of the LDA. Multilayer perceptron (MLP) networks can be used for learning the nonlinear mapping from the input features to the phone class identifiers. The number of the output layer nodes corresponds to the number of the phone classes and the training of the network is supervised. The activation values of the output nodes of the MLP network are then used as the values of the nonlinear discriminant functions for separating classes.

The number of the input nodes in the MLP corresponds to the dimension of the original (context) feature vector D. If the number of the nodes in the output layer is larger than the desired output feature D', the dimension can be reduced by KLT (this may be beneficial also because of the decorrelation effect). The number of the nodes in the hidden layers can be arbitrary. Also the nonlinear activation functions of the nodes can be arbitrary. This is the most flexible class of

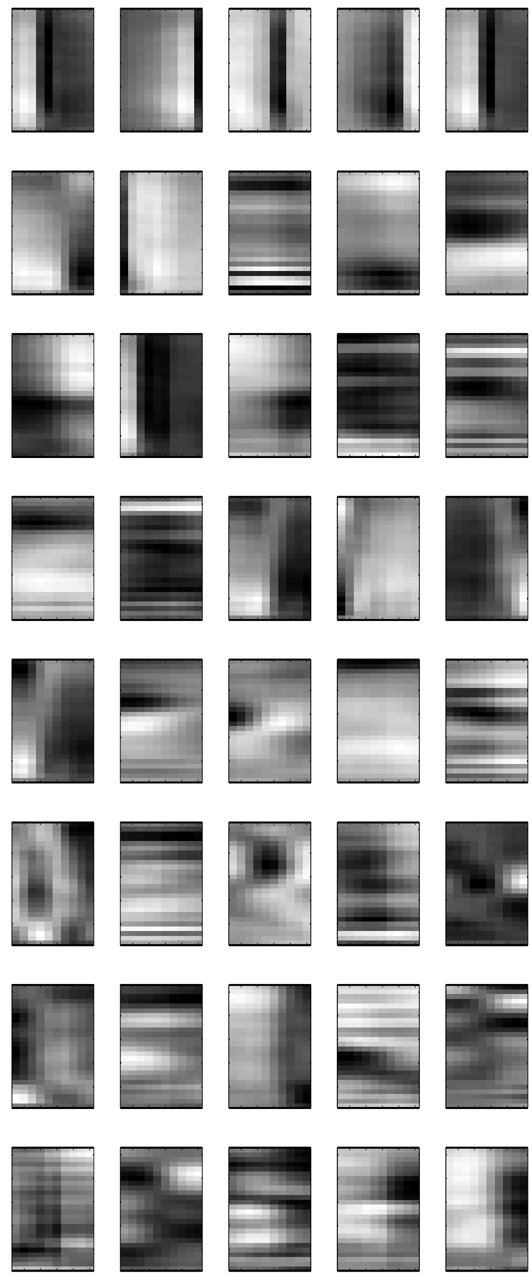

Fig. 1. ICA basis vectors for ten-frame logmel-spectrum windows. Each subimage corresponds to one column of the mixing matrix. Vertical axis of each subimage corresponds to the mel-channel and horizontal axis corresponds to the time frame. The dimension of the original context feature vector was first reduced to 40 by PCA and after that the unsupervised FastICA algorithm (MATLAB Toolkit, [2]) was applied. Some basis vectors are purely temporal or purely spectral edge filters while some of them have been tuned to detect more complex spectro-temporal patterns.

transformations (in principle including all previous transformations as special cases). It has been proved that MLPs are universal approximators of nonlinear functions. However, this flexibility can also cause problems. If the number of the free parameters in the network is too large compared to the complexity of the training data the network may overlearn the training data and does not generalize well. However, there are several ways for controlling the learning and the generalization performance; one simple but efficient method is to use early stopping criterion: the training is stopped when the error of the independent validation set does not decrease in further iterations.

MLPs have been been shown to give promising results in the feature transformation, see earlier work e.g. [5].

3. EXPERIMENTS

In order to compare different feature transformations, phone recognition experiment was carried out using the TIMIT database. The training set consisted of all si and sx sentences from the 496 speakers of the original training set (3698 sentences) and the test set consisted of all si and sx sentences from the complete 168-speaker test set (1344 sentences).

The original phoneset consists of 61 phone classes. Some of these classes are highly overlapping, e.g., there are 10 separate classes for various kinds of silences (beginning and ending mark of speech '#h', pause 'pau', 'epi', closures 'bcl', 'dcl', 'gcl', 'pcl', 'kcl, 'tcl', and glottal stop 'q'). For this reason some authors have reduced the number of the classes of the original phoneset. In this work the class merging was done according to [6] resulting in 39 phone classes. According to [6], merging the closures had the major impact in the recognition performance, but further merging of the allophones led only to minor improvements.

Each phone was modeled by a three-state left-to-right HMM and each state was modeled by a mixture of diagonal Gaussians. The models were trained using the HTK software [7]. The models were initialized using 'flat start' method, i.e., no phone segmentation information was used. Each state contained only one Gaussian in the beginning of the training. The number of the Gaussians was then expanded into 2, 4, 8, and 16, after each expansion running one cycle of the BW re-estimation. Finally, five cycles of the Baum-Welch training were performed using the 16-component mixtures. This training scheme was applied indetically to each feature transformation. In the recognition, the back-off bigram model was used for phones (computed from the TIMIT sentences which were present in the training set). Language model weights were determined separately for each feature set by using 370 sentences of the training set. For MFCC, PCA, ICA, and LDA features, the weight was 3.0 and for the 24-component NLDA feature it was 4.0.

All features were computed from the five-frame logarithmic mel-spectrum windows. The mel-spectrum vectors were computed from 25 ms Hamming-windowed speech frames at every 10 ms interval. The number of the mel-channels in each frame was 24. The dimension of the five-frame context window D was thus 120. The baseline feature vector consisted of 12 MFCC coefficients (utterance-wise cepstral mean subtraction) with the first-order deltas resulting in a 24-component feature vector. The deltas were computed from the five-frame windows. The number of the output feature vector components was fixed to be the same in all feature transformations ($D' = 24$). In the following feature computations, utterance-wise mean was subtracted from the logarithmic mel-spectrum vectors.

PCA and ICA bases were formed using middle parts of the phone segments. This was only for reducing the number of the frames, no class information was utilized when forming the feature transform matrices. Fast fixed-point algorithm was used for computing the ICA basis [2].

It is interesting to compare the DCT against PCA. However, it is not reasonable to use one-dimensional DCT to the concatenated feature vectors. Instead, a two-dimensional DCT can be applied to the successive mel-spectrum frames. First, one-dimensional DCT is performed for individual mel-spectrum vectors resulting in the conventional MFCC vectors. After that another one-dimensional DCT is applied to the successive MFCC vectors component-wise over time. However, the feature vectors projected to the three first basis vectors of the DCT correspond to the average, delta, and delta-delta features which is the conventional MFCC feature set. Here PCA clearly outperformed the combination of the MFC vector with the first-order deltas.

NLDA-features were obtained by training an MLP network for discriminating phone classes. The number of the hidden layer nodes (600) was set to be five times the number of the input nodes (120). The number of the output layer nodes was the size of the phone set (39). MLP training was done using the ICSI software. Softmax-activation function was used in the output layer during the training, but when using the MLP output as a feature vector to HMM, this nonlinearity was removed (in order to get more Gaussian distributed features for a mixture-of-Gaussians HMM [5]). The 39-dimensional output vector was reduced to 24 components by KLT (transformation matrix obtained from the training data). Before KLT, frame-wise mean was subtracted from each phone class posterior vector in order to eliminate the bias term of the MLP [8].

The results of the experiments are in Fig. 2. All experimented feature transformations outperformed the baseline MFCC features. It was also experimented to concatenate PCA, ICA, LDA, and MLP features and then decorrelate and reduce the dimensionality of the resulting vector

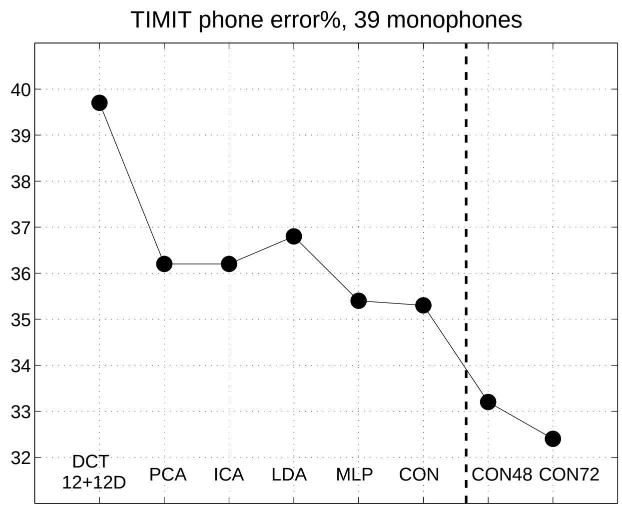

Fig. 2. TIMIT phone recognition using different feature transformations. All features were computed from five-frame logarithmic mel-spectrum windows (120 feature vector components). DCT12+12D is the MFCC with the first-order deltas which was the baseline feature. All feature vectors on the left side of the vertical dash line had 24 components. CON denotes concatenated feature vector (PCA, ICA, LDA, MLP) followed by KLT, CON48 contained 48 feature components and CON72 72 components, respectively.

by KLT. This gave further improvement to the recognition accuracy.

4. DISCUSSION

The main purpose of the experiments presented in this work was to compare different feature transformations. Therefore, simple 3-state context-independent phone HMMs were used. The results of all feature sets could be improved by using more detailed acoustic models, e.g. context-dependent phones. Also, better overall results would be obtained by using larger context than five frames. This was confirmed in a preliminary experiment.

Another topic is the use of Gaussian mixtures. It may favor certain kinds of features. Therefore, the same features could also be compared using another classifier, e.g. an MLP-based HMM.

Finally, in this work only global transformations were considered. However, the transformation could also be class-specific, or in case of HMMs, state-specific. Integrating the feature transformation more closely into the classifier could help in detecting more class-specific cues from the input.

5. CONCLUSIONS

In this work, four feature transformations were experimented in the TIMIT phone recognition task. The input feature vector consisted of the five consecutive frames of the logarithmic mel spectra. Two of the transformations were unsupervised (PCA and ICA), i.e., no class information of the frames were used when forming the transformation matrices, and two of the transformations were discriminative (LDA and NLDA). All four feature transformations clearly outperformed the baseline feature which consisted of the MFC coefficients with the first-order deltas.

It was interesting that the unsupervised PCA and ICA performed as well as the discriminative LDA-based features. The differences between the PCA and ICA features were not visible using HMMs with Gaussian mixtures, but some other classifier might benefit more from the ICA. PCA removes only the second-order dependencies between the feature vector components while ICA removes also higher-order dependencies.

The best results were obtained when the feature vectors from all four transformations were concatenated together and the resulting feature vector was decorrelated by KLT. This suggests that the different features contain complementary information. Future work will be done on determining what is the best way to combine the different features. Instead of concatenating the feature vectors, separate recognizers can be run in parallel, and the model combination can then be done in the state-likelihood level.

6. REFERENCES

[1] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford, 1995.

[2] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley & Sons, 2001, http://www.cis.hut.fi /projects/ica/fastica/index.shtml.

[3] I. Potamitis, N. Fakotakis, and G. Kokkinakis, "Spectral and cepstral projection bases constructed by independent component analysis," in *Proceedings of the ICSLP*, 2000, vol. 3, pp. 63–66.

[4] S. Kajarekar, B. Yegnanarayana, and H. Hermansky, "A study of two dimensional linear discriminats for ASR," in *Proceedings of the ICASSP*, 2001, vol. 1.

[5] H. Hermansky, D. Ellis, and S. Sharma, "Tandem connectionist feature stream extraction for conventional HMM systems," in *Proceedings of the ICASSP*, 2000, vol. 3, pp. 1635–1638.

[6] K. Lee and H. Hon, "Speaker-independent phone recognition using hidden Markov models," *IEEE Tr. ASSP*, vol. 37, no. 11, pp. 1641–1648, 1989.

[7] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *The Hidden Markov model Toolkit, version 3.1*, 2001, http://htk.eng.cam.ac.uk/.

[8] S. Sharma, *unpublished work*, 2002.