AUDIO SEGMENTATION BY FEATURE-SPACE CLUSTERING USING LINEAR DISCRIMINANT ANALYSIS AND DYNAMIC PROGRAMMING

Michael M. Goodwin and Jean Laroche

Creative Advanced Technology Center
1500 Green Hills Road, Suite 205
Scotts Valley, CA 95066
mgoodwin,jeanl@atc.creative.com

ABSTRACT

We consider the problem of segmenting an audio signal into characteristic regions based on feature-set similarities. In the proposed method, a feature-space representation of the signal is generated; then, sequences of feature-space samples are aggregated into clusters corresponding to distinct signal regions. The clustering of feature sets is improved via linear discriminant analysis (LDA); dynamic programming (DP) is used to derive optimal cluster boundaries. The method avoids the heuristics employed in various feature-space segmentation schemes and is able to derive an optimal segmentation once the LDA and DP cost metrics have been chosen. We demonstrate that the method outperforms typical feature-space approaches described in the literature. We focus on an illustrative example of the basic segmentation task; however, by judicious design of the feature set, the training set, and the dynamic program, the method can be tailored for various applications such as speech / music discrimination, segmentation of audio streams for smart transport, or song structure analysis for thumbnailing.

1. INTRODUCTION

Segmentation of audio signals into meaningful regions is an essential aspect of many applications, for instance speech / music discrimination for broadcast transcription, audio coding using transient detection and window switching, identification of suitable audio thumbnails, and demarcation of songs in continuous streams for database creation or smart transport. To perform effectively, such applications rely on a basic signal understanding provided by automatic segmentation.

Segmentation approaches can be loosely grouped into statistical methods and methods based on feature-space distance metrics. In this paper we are primarily concerned with the latter; we present a method that relies similarly on intuitive signal features but which avoids the heuristics typically called for in feature-analysis methods. We do not directly explore comparisons with primarily statistical methods, but the proposed algorithm can be extended appropriately for comparison to Gaussian-mixture modeling methods or other statistically driven approaches [1, 2].

In the proposed algorithm, the segmentation task is interpreted as a feature-space clustering problem. We show that typical feature-space segmentation schemes can be improved by the use of judicious data transformations. We further describe the application of dynamic programming to derive robust segmentation points, which correspond to cluster transitions in the problem framework. The proposed method includes the use of intuitive signal features as a front end, statistical considerations to improve discrimination between feature sets, and optimal estimation of segment boundaries. The resulting segmentation is optimal given the feature set, a training set for the discrimination stage, and the design of the cost functions for the dynamic program; we show that the dynamic program can be designed to achieve functional objectives, meaning that the proposed algorithm derives a robust segmentation without relying on any arbitrary heuristics or thresholds.

2. FEATURE SPACE

An audio signal can be represented in a feature space by carrying out a sliding-window analysis to extract feature sets on a frame-to-frame basis. Examples of such features include zero-crossing rate, spectral centroid, tilt, and flux, and so on [3, 4]. In such a scheme, each window hop yields a new set of features; for the i-th frame:

$$w[n]x[n+iL] \rightarrow \boxed{\text{Feature analysis}} \rightarrow f_i\{x\} \qquad (1)$$

The output of the feature analysis block is the feature vector $f_i\{x\}$, which will be referred to hereafter simply as a column vector f_i.

The sequence of feature vectors f_i provides a feature-space representation of the input signal. From this representation a variety of similarity (dissimilarity) metrics can be computed for successive feature vectors, for instance a a vector difference norm

$$d_{ij} = (f_i - f_j)^H D^H D (f_i - f_j), \qquad (2)$$

where $D^H D$ is the identity matrix for the Euclidean distance, the inverse covariance matrix of the feature set for the Mahalanobis distance, or some other feature weighting specific to the particular distance measure. For any such metric, the sequence of differences between successive feature vectors is a *novelty function* which quantifies the extent of change in the audio signal between adjacent frames [5]. Feature-based segmentation schemes reported in the literature typically use such an approach to determine segment boundaries: peaks in the novelty function indicate boundaries in the audio signal, *i.e.* there is a change if successive features f_i and f_{i+1} are deemed dissimilar enough [4, 5].

In order to make the decision as to whether successive frames are substantially dissimilar to indicate a segmentation boundary, a heuristic threshold for this determination must be established. Consider the typical novelty functions shown in Fig. 1, which were derived from a sample audio stream based on a typical feature set. Plot 1(a) uses the Euclidean distance, which simply determines the feature-space distance using the raw feature values; plot 1(b) uses the Mahalanobis distance, in which the features are scaled such that each contributes equally to the distance measure. Noting the

0-7803-7850-4/03/$17.00 © 2003 IEEE

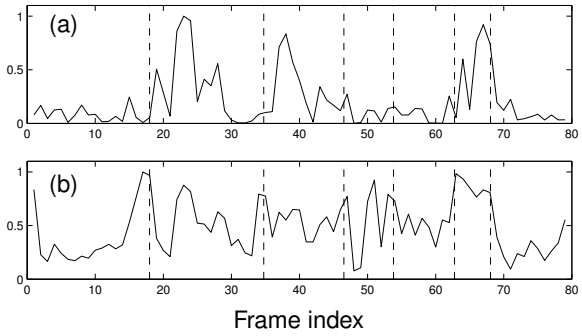

Figure 1: Normalized novelty functions based on (a) Euclidean distance and (b) Mahalanobis distance. The dashed lines indicate the actual segment boundaries.

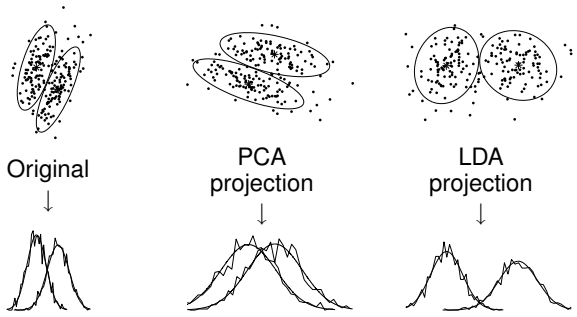

Figure 2: A two-class example of PCA and LDA transformations. The LDA projection is more effective for discriminating between the classes than the PCA projection.

actual segment boundaries indicated in the plots, we see that these novelty functions tend to have peaks in the vicinity of the boundaries but that they also have significant spurious peaks. Setting a decision threshold for peak-picking, then, is an *ad hoc* procedure; the key shortcoming of such methods is that the robustness or optimality of a given threshold cannot be guaranteed.

In the following sections, we explore two steps to enhance the performance of feature-space segmentation: first, we show an improved novelty function in which the peaks corresponding to desired points-of-change are accentuated while spurious peaks are suppressed; and second, we propose a more robust strategy than peak-picking such that heuristic thresholding is avoided altogether in the determination of segment boundaries.

3. CLASSIFICATION AND CLUSTERING

The problems described above can be effectively addressed by casting the segmentation task as a clustering problem. In this section we describe the basic framework of classification and clustering, explain how to interpret segmentation from a clustering perspective, and describe the application of well-known classification techniques to improve segmentation performance.

3.1. The classification problem

In the basic classification problem, there is an initial data set for which the class of each sample is known. Given a new sample, then, the question is to which class it should be assigned. Such assignments are made by establishing and applying a decision rule; for instance, a simple rule could be that a new sample is assigned to the class whose mean it is closest to in feature space. The details of decision rules are beyond the scope of this paper; the pattern recognition literature is rife with references on this topic [6, 7]. A related question which is of central importance is whether the raw feature data can be transformed or projected into a new feature space in which the classes are easier to distinguish and a more robust decision rule can be found. We can indeed discuss such transforms without considering the specifics of the decision rule.

3.2. Linear discriminant analysis

Linear discriminant analysis (LDA) is one technique for transforming raw data into a new feature space in which classification can be carried out more robustly. Given a training set consisting of raw feature data $\{f_i\}$ and a known class for each sample, the idea in LDA is to find a matrix P such that for the new data $a_i = Pf_i$ the separation between classes is maximized while the variance within a class is minimized. There are several variants of LDA [7]; in one commonly used variant, the optimal P consists of the eigenvectors of a generalized singular value decomposition involving the scatter matrices of the training set. In general the dimensionality of the transformed feature set is one less than the number of training classes; the dimensionality can be further reduced, however, by incorporating in P only those eigenvectors corresponding to the largest singular values determined in the scatter SVD.

An example of LDA is shown in Fig. 2. The original data consists of two classes, and the task is to find a projection onto one dimension which separates the classes. The first example shows a projection of the data onto the original horizontal coordinate; class separation is not achieved. The second shows the application of principal component analysis (PCA), which often exhibits poor performance in classification problems since the principal axes do not necessarily entail discriminatory features [7]. The third example shows LDA achieving a significant improvement in class separation; note that the LDA transformation separates the class means while attempting to sphere the data classes.

3.3. Segmentation as clustering

In the feature-space framework, an audio signal is equivalent to a sequence of feature vectors. The task of segmentation can thus be interpreted as a classification problem in that it corresponds to grouping subsequences of these feature vectors into segmentation regions. If the goal is to assign each feature vector to an *a priori* class established in a learning stage, then indeed this is a *classification* problem. More generally, however, the segmentation task amounts to observing a sequence of arbitrary feature samples and aggregating subsequences into classes in some fashion. Such aggregation and the inherent generation of classes based on observed data is called *clustering*.

Given an audio stream with perceptually distinct regions, raw feature vectors consisting of pitch, zero-crossing rate, spectral tilt, *etc.* will not necessarily be clustered in feature space according to the various regions. Noting the spurious peaks in the novelty functions of Fig. 1, it is clear that feature vectors for the same region can actually be far apart in feature space. The clustering of regional features can be substantially improved by applying an LDA transformation derived from a representative training set. For example, suppose an LDA matrix is derived to discriminate between the classes in a given set and that a signal is constructed by draw-

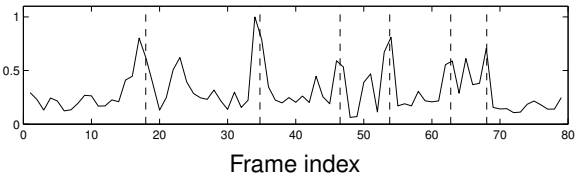

Figure 3: Improved novelty function based on Euclidean distance after an LDA transform. The dashed lines are the actual segments.

ing excerpts from those classes. A novelty function based on the raw features may be problematic for segmentation. On the other hand, since cluster separation is increased by LDA, the novelty of the LDA features exhibits stronger peaking at the segment boundaries as shown in Fig. 3; furthermore, the reduction of within-class variance suppresses the spurious peaks within segments. For a signal not composed of samples drawn from the LDA training set, the LDA will still help to improve the feature-space clustering if the training set is appropriate for the signal in question. The LDA training finds a transform of raw feature data into a feature space which best matches the perceptual discrimination defined by the examples in the training set; as long as similar perceptual discrimination applies, effective performance of the LDA can be expected.

A related explanation of the advantage of LDA is that in the novelty functions of Fig. 1 the distance function uses a weighting (Euclidean or Mahalobnis) which is uninformed of the relative importance of the various raw features for cluster discrimination. In this light, transforming the data via LDA is equivalent to using an optimal feature weighting for the distance metric in Eq. (2).

4. CLUSTERING OF SEQUENTIAL DATA USING DYNAMIC PROGRAMMING

In this section we first discuss the shortcomings of the novelty peak-picking paradigm for segmentation. We then propose a more robust approach to clustering based on dynamic programming (DP).

4.1. Shortcomings of the novelty function

As previously described, the raw novelty functions of Fig. 1 exhibit spurious peaks which will degrade the performance of novelty-based segmentation schemes. It is clear in Fig. 3 that LDA improves the situation, but that such peaks still occur. The reason for this is a fundamental shortcoming of the novelty function: it is designed to identify local changes between samples; global changes between groups of samples, however, are of greater importance to the segmentation task. LDA does not resolve this issue entirely since it is not at all designed to enhance inter-sample novelty but moreso inter-cluster novelty. The spurious peaks arise when two successive samples within a single cluster are further apart than successive samples in different clusters, which is a common occurrence in tightly packed feature spaces – even after a clustering transformation such as LDA. A more degenerate case can also be envisioned; if a sequence of samples progresses gradually across a cluster and into the adjacent cluster, the novelty function may not exhibit any peaks at all even though a cluster transition has occurred. What is needed instead of a local novelty measure, then, is a detection of global signal trends. In the following section, we describe a dynamic program which essentially looks for the means of subsequences and identifies segmentation boundaries when the samples start to aggregate around a new mean.

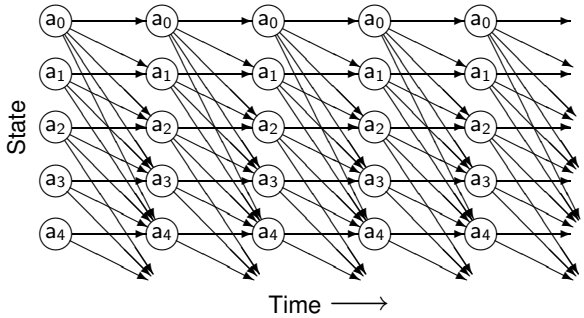

Figure 4: Partial state transition diagram of the dynamic program for feature-space clustering. The label corresponds to the feature vector associated with the state.

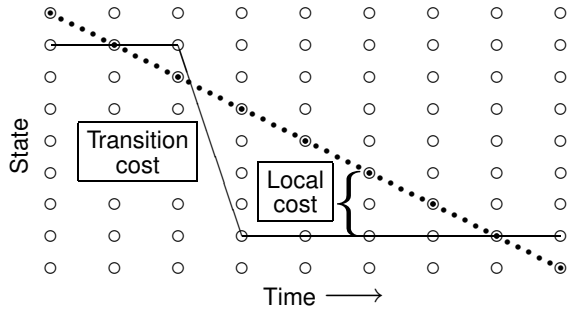

Figure 5: In the dynamic program for feature-space clustering, the diagonal (dotted) is the nominal feature path. A candidate cluster path with one transition is shown; there is a transition cost as well as a local cost for being in any state that is not on the nominal path.

4.2. Structure of the dynamic program

Given the LDA-transformed feature sequence $\{a_i\}$, which will exhibit better clustering than the raw data $\{f_i\}$, dynamic programming can be used to find cluster transitions. Assuming there are N feature sets in the sequence, an $N \times N$ state machine such as that in Fig. 4 is constructed. For each time frame j, there are N candidate states; letting i be the vertical state index, each state S_{ij} is associated to the feature vector a_i as shown in the figure.

The diagonal path of the state transition diagram in Fig. 4 corresponds to the nominal feature-space trajectory of the signal: at time j, the nominal path is in state j. Recall that DP is able to find a path through the state diagram which optimizes some specified cost function which can be composed of a local cost for each state as well as costs for transitions between states. The objective of the dynamic program design, then, is to select these cost functions such that the resulting optimal path indicates not this nominal feature-space trajectory but rather a trajectory through clusters. This cluster path will be a stepwise traversal of the state transition diagram. Each plateau corresponds to a cluster and each step is a transition between clusters; the feature vector for a cluster plateau is the characteristic feature set for that cluster. The nominal feature path and a candidate cluster path are depicted in Fig. 5.

4.3. Cost functions for the dynamic program

In the dynamic program, local and transition cost functions must be designed to achieve the desired segmentation task. The local cost should reflect how likely it is to be in state i at time j. Recall

that we want the final trajectory to be composed of horizontal segments separated by transitions. In any horizontal segment of the path, we remain in the same state i_0: the local cost of state i_0 at time j ($S_{i_0 j}$) should be small if the feature vector measured in the signal at time j (the one associated to state S_{jj}) is similar to the feature vector associated to state i_0, and high otherwise. This will ensure that along a horizontal segment of the path, the successive feature vectors extracted from the signal are similar to the feature vector a_{i_0} that represents that horizontal segment. An intuitively reasonable choice for the cost function is thus the Euclidean distance $||a_i - a_j||$ so that the cost of being in state S_{ij} is the distance between the feature vectors of that state and of the diagonal state S_{jj} for that time index. The states between which the local cost distance is measured are indicated in Fig. 5.

The aggregate local cost for a candidate path is the sum of the local costs for the states in the path. For the Euclidean cost metric, this is clearly zero for the nominal diagonal path. Assuming for the moment, however, that the transition cost is infinite such that a horizontal path must be chosen, it can be shown that the choice of Euclidean distance is actually optimal. Considering a set of N feature vectors and letting a_m be the single feature vector of the chosen path, the aggregate local cost for the path is

$$L(a_m) = \sum_{j=0}^{N-1} (a_m - a_j)^H (a_m - a_j), \quad (3)$$

which is minimized if a_m is the mean of the set; a_m must however be chosen from the sample set. To find the best choice, we write:

$$L(a_m) = \sum_{j=0}^{N-1} ||a_m - \bar{a} + \bar{a} - a_j||^2 \quad (4)$$

$$= N||a_m - \bar{a}||^2 + \sum_{j=0}^{N-1} ||a_j - \bar{a}||^2, \quad (5)$$

where the cross-terms in the expansion of Eq. (4) cancel since $\bar{a}$ is the mean of the set. Noting that the second term in Eq. (5) is not dependent on a_m, we see from the first term that the optimal choice is the set member closest to the mean. Thus, the optimal horizontal path is the path which stays in the state whose feature vector is closest to the mean of the set. In the clustering framework, this feature is the closest member of the cluster to the cluster mean and is the optimal choice to be a representative of the cluster.

The transition cost does not admit to direct formulation as readily as the local cost, but several constraints are clear. First, a high cost should be associated to switching from state i to state j if the corresponding feature vectors are similar; however, there should be zero cost for a transition from i to i (since we are looking for horizontal paths). Conversely, the cost should be small for a transition between very dissimilar feature vectors (so real transitions in the audio are not missed). An intuitive choice for the transition cost between two states is then the inverse of the Euclidian distance between the corresponding feature vectors; a constant cost can also be added for any non-horizontal transition to further favor clustering into horizontal segments. Note that for the segmentation task, only horizontal or downward transitions are allowed in the DP; upward transitions could of course be introduced for some alternate applications.

The robustness of the proposed LDA-DP segmentation by clustering is indicated in Fig. 6; audio demonstrations will be given at the conference and are available online [8].

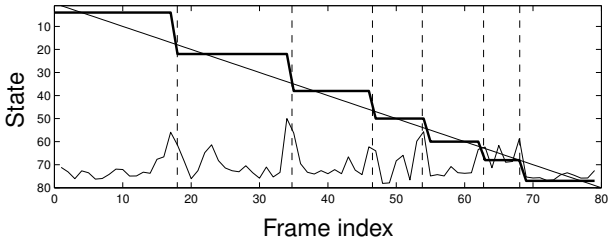

Figure 6: The cluster path (bold) derived by the dynamic program exhibits transitions which match the actual segment boundaries (dashed). The nominal diagonal path is shown along with the LDA novelty function from Fig. 3.

5. CONCLUSIONS AND FUTURE WORK

We have described an audio segmentation algorithm wherein features are extracted from the signal, transformed via LDA to optimize cluster scattering, and then clustered using a dynamic program. The LDA-DP routine essentially converts a feature-space trajectory into a cluster-space trajectory wherein cluster transitions indicate points of significant global change in the signal. The system is general and can be tailored for various applications by appropriate selection of the signal feature set, training set, cost functions, and DP structure. Once the system has been designed, it is able to find an optimal segmentation without relying on heuristic metrics as other methods do.

Possibilities for future work include theoretical areas such as kernel-based clustering approaches, backpropagation for training of the DP cost functions, and formal comparison with other approaches such as the alternative hidden Markov modeling scheme described in [2]. Also, it is of interest to address practical aspects such as validating the segmentation via human agreement as in [4].

6. REFERENCES

[1] E. Scheirer and M. Slaney, "Construction and evaluation of a robust multifeature speech/music discriminator," *Proceedings of IEEE-ICASSP*, vol. 2, pp. 1331–1334, April 1997.

[2] J. Aucouturier and M. Sandler, "Segmentation of musical signals using hidden Markov models," *AES 110th Convention*, May 2001, preprint 5379.

[3] E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-based classification, search and retrieval of audio," *IEEE Multimedia*, vol. 3, no. 2, pp. 27–36, Fall 1996.

[4] G. Tzanetakis and P. Cook, "Multifeature audio segmentation for browsing and annotation," *Proceedings of IEEE-WASPAA*, pp. 103–106, October 1999.

[5] J. Foote, "Automatic audio segmentation using a measure of audio novelty," *Proceedings of IEEE-ICME*, vol. I, pp. 452–455, July 2000.

[6] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley and Sons, 1973.

[7] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. San Diego, CA: Academic Press, 1990.

[8] M. Goodwin, "Demo of audio segmentation by clustering," www.atc.creative.com/users/mgoodwin/segment.html.