Feature Fusion and Redundancy Pruning for Rush Video Summarization

Jim Kleban, Anindya Sarkar, Emily Moxley, Stephen Mangiat, Swapna Joshi, Thomas Kuo and B.S. Manjunath

Vision Research Laboratory, University of California, Santa Barbara, USA
kleban,anindya,emoxley,smangiat,sjoshi,thekuo,manj@ece.ucsb.edu

ABSTRACT

This paper presents a video summarization technique for rushes that employs high-level feature fusion to identify segments for inclusion. It aims to capture distinct video events using a variety of features: k-means based weighting, speech, camera motion, significant differences in HSV colorspace, and a dynamic time warping (DTW) based feature that suppresses repeated scenes. The feature functions are used to drive a weighted k-means based clustering to identify visually distinct, important segments that constitute the final summary. The optimal weights corresponding to the individual features are obtained using a gradient descent algorithm that maximizes the recall of ground truth events from representative training videos. Analysis reveals a lengthy computation time but high quality results (60% average recall over 42 test videos) as based on manually-judged inclusion of distinct shots. The summaries were judged relatively easy to view and had an average amount of redundancy.

Categories and Subject Descriptors

I.2.10 [**Vision and Scene Understanding**]: Video analysis

General Terms

Algorithms, Experimentation, Performance

Keywords

Feature Fusion, Video Summarization, Dynamic Time Warping, SIFT-based Object Detection

1. INTRODUCTION

Digital production now dominates the video industry from amateur videos for the Internet captured on inexpensive digital cameras to professional television and motion picture productions that are digitized for editing. This has created a database that is too large to manually evaluate, but can be automatically evaluated by a computer. Retrieval can be facilitated immensely using compact video summaries instead of the entire videos. Much work has been done with content-based approaches [5] for creating summarizations for movies [6] and other types of video such as sports[4] and news[10]. However, the task of creating such summaries for unedited b-roll, or rush, films that contain many repeated director takes of a scene is relatively new.

This paper focuses on the problem of creating a compact video summary where visually dissimilar scenes are identified. Candidate segments containing salient camera motion, a high level of speech activity, or rapidly changing visual features are emphasized for inclusion, inspired by the user-attention model described in [9]. Video redundancies (shot retakes) and irrelevancies (color bars and objects like clapboards) are removed such that summaries contain objects and events such as those described in the NIST-provided ground truth annotations. This work does not attempt to semantically label the distinct scenes, which provides scope for future work if the summaries are to be used for browsing results of a search query.

The video summarization algorithm was used to summarize rushes from BBC programming as selected for the TRECVID benchmarking collaboration [12]. The rushes contain unedited footage of BBC dramas including shot setup and retakes. The system is explained in section 2, followed by results in section 3, and analysis in section 4.

2. SYSTEM WORKFLOW

The system incorporates high-level feature fusion of concepts important for redundancy removal and distinct shot inclusion. The features are fused as a weighted combination, where the weights are optimally estimated to maximize the recall of manually-annotated key events provided by collaborators in the development videos. Irrelevant object removal is performed on the sample frames to remove known trivialities such as color bars, monochromatic frames, and clapboards. A final k-means clustering selects visually distinct frames to create the summarized video. Figure 1 depicts the system overview.

2.1 Preprocessing

A variety of preprocessing steps are performed to extract frames, determine shot boundaries, and generate low-level features characterizing the distinct frames. The rush data were MPEG-1 encoded at 25 frames per second. Extraction

TVS'07, September 28, 2007, Augsburg, Bavaria, Germany.

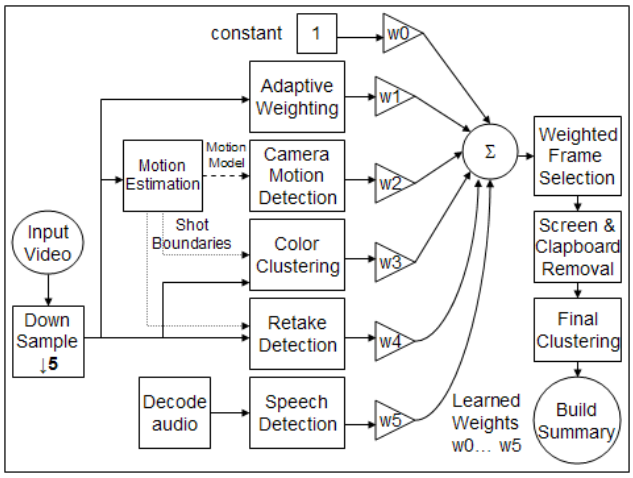

Figure 1: System Flow Diagram. Final output summaries are 4% of the original video length.

of high-quality video frames leads to a more pleasing final video but decoding time increases by 10x. A subsampling factor of five frames was used for these experiments.

2.2 Feature Extraction

Analysis of the ground truths provided for development data revealed that the important sections to be included in summarized video were of four types: shots containing camera motion, shots of people entering or leaving a scene, shots showing certain objects, and shots of distinct events. Since high-level features can be indicators of the relative importance of a particular video segment [9], appropriate features were extracted to capture these four types. Additionally, repeated takes of the same scene are irrelevant and should be excluded from the summary. Five high-level features are extracted from the original video:

i) a k-means-based feature that weights distinct scenes,
ii) a camera feature that weights salient camera motions such as panning and zooming,
iii) an acoustic feature that weights segments with speech,
iv) an adaptive sampling function that weights rapidly changing visual segments in the HSV colorspace, and
v) a DTW-based feature that weights unrepeated segments and the longest of repeated segments.

Intuitively, feature (i) weights shots showing objects and distinct backgrounds, (ii) weights shots with camera motion,(iv) weights shots of people entering or leaving a scene and events.

2.2.1 K-Means Clustering

A repeated clustering technique is used to include visually distinct portions of the video in the summary. The main steps of the method are as follows.

1) The video is segmented based on a global 2D affine motion model [2]. Significant changes in the model parameters between a pair of consecutive frames denote a shot boundary. A low threshold is used to oversegment the video. Short segments less than 40 frames long are merged. One keyframe is extracted from the center of each segment.
2) K-means is then employed on the HSV color features of the keyframes. The number of clusters is set to two-thirds of the number of input segments. The keyframe closest to each cluster centroid is selected.
3) Selected keyframes are convolved with a 90-frame Hamming window.
4) K-means clustering is repeated five times with different initial centroid locations, and the smoothed selections are summed.
5) After normalization, the result is a function that weights areas often selected by clustering for inclusion in the final summary.

2.2.2 Camera Motion (Pan/Zoom/Tilt)

The ground truth examples also included instances of camera motion events such as panning and zooming. Affine motion based parameters (explained in detail in [2]) can identify these camera motions. Pans, tilts, and zooms are identified and smoothed with an 80-frame averaging filter to alleviate the effect of sporadic false positives. The final result is an output feature weights areas with panning, tilting, and zooming for inclusion in the summary. The problem of distinguishing between *significant* camera motions occurring within a scene, and motion during camera setup and shot reset is unresolved.

2.2.3 Speech Based Feature

For the TRECVID rushes task, which focused on summarizing drama footage, audio segmentation can be helpful in supplementing visual feature analysis. Speech segments with high energy often occur when an actor is speaking on-camera during a scene. Offscreen chatter may also be classified as speech, but it would presumably be at a lower level. Therefore, the main use of speech to assist summarization is to help remove segments in-between takes or black frames and color bars.

The system uses three acoustic features outlined in [8] to distinguish speech and non-speech segments: High Zero-Crossing Rate Ratio (HZCRR), Low Short-Term Energy Ratio (LSTER), and Spectrum Flux. The ones used here can be calculated quickly for very large videos and can adequately perform simple speech/non-speech classification. Speech is characterized by intermittent silence and voiced sounds with strong spectral components. Therefore it has high HZCRR, LSTER, and spectrum flux, making it distinguishable from environment noise.

The system uses these acoustic features to discriminate between environment, pure tone (often associated with color bars) and speech. A training set was used to determine centroids for nearest neighbor clustering. Using a similar test set, approximately 78% of speech and 89% of environmental noise were correctly classified. In addition to classification, RMS energy was calculated for each one-second window within a video. Only the "loudest" 30% of the speech segments are kept. Thus, the output of the audio classifier, as can be seen in Fig. 4(c), is a binary speech/no-speech vector. In this way, high RMS speech was used to correlate with "important" events within the rushes videos.

2.2.4 Adaptive Sampling in HSV Colorspace

The idea behind adaptive sampling is to sample more frequently during scenes with more action or varied content. More keyframes are allocated to segments that change

quickly in the HSV colorspace. An L_2 distance in 12 dimensional HSV colorspace between adjacent frames is used as a measure of scene change.

Since this raw difference exhibits significant noise, a 25-point median filter was chosen to compensate. A median filter also disregards abrupt cuts in the shot. A window of 25 frames was chosen since it represents one second, the subjective perceptual limit.

2.2.5 *DTW-Based Redundancy Removal*

Since many retakes are present in the rushes data set, the task requires an effective redundancy removal system for the summary. Once the sub-shot segment boundaries are detected using the aforementioned motion-based detection, the segments are enumerated into three types:
(a) unique and without repeat,
(b) repeated and the longest of all similar segments, and
(c) repeated but not the longest of all similar segments.

Ideally, (a) and (b) are to be retained, with the assumption that (b) segments are more informative than (c) segments. This assumption is logical because (c) segments may be missing parts of the scene if a shot is cut short due to an actor or director mistake. An output feature scales these segments higher or lower depending on whether they are to be retained or discarded.

For comparing segments of dissimilar lengths for repeat detection, the system uses an approach based on dynamic time-warping (DTW) [13]. DTW is a time-normalization method used for comparing sequences of dissimilar lengths. For DTW-based distance computation, each frame is represented by a 1125-dimensional localized color histogram [3]. Figure 2 shows a matrix of the DTW-based distance between hand-segmented shots in a video. Segments that reflect a retake of the same scene exhibit a low DTW-based distance, while distinct shots maintain a high distance. A threshold of 0.41 was empirically chosen to flag repeated scenes, using the receiver operating characteristic (ROC) depicted in Figure 3.

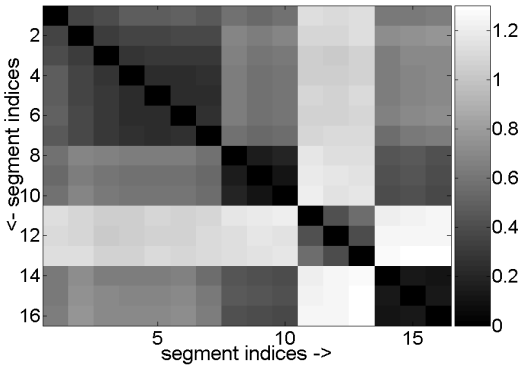

Figure 2: Pairwise DTW distance between manually segmented video shots. The video has 4 sets of segment repeats: 1-7, 8-10, 11-13, and 14-16.

The DTW algorithm is as follows. Let there be M segments $\{s_i\}_{i=1}^{M}$ in the video, obtained after shot boundary detection. After computation of the DTW-based distance [13], we have an $M \times M$ inter-segment distance matrix. Let $d(s_i, s_j)$ denote the DTW distance between the i^{th} and j^{th}

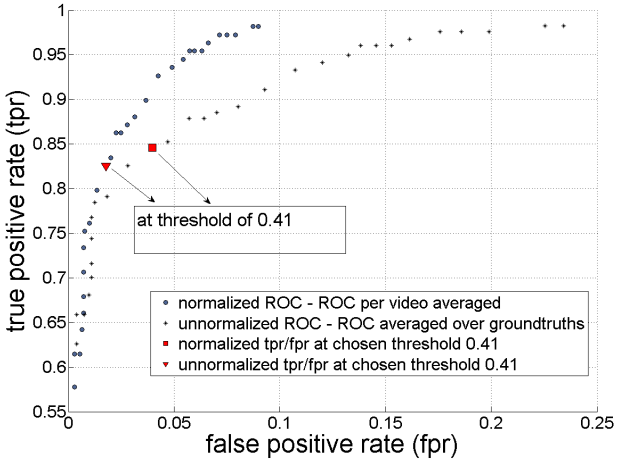

Figure 3: Receiver Operating Characteristic for varying thresholds of DTW. Empirical threshold was set at the elbow of 0.41.

segments. Often in videos the search space can be pruned to adjacent segments, making it necessary to compute only $(M-1)$ distances. However, in the general case distances between all pairs of segments need to be computed. Two segments, s_i and s_j, are considered similar if $d(s_i, s_j) < \delta_{th}$, where δ_{th} is a threshold in the localized color histogram space.

An issue with the similarity between two segments is that the following scenario can occur, $d(s_i, s_j) < \delta_{th}$ & $d(s_j, s_k) < \delta_{th}$ but $d(s_i, s_k) > \delta_{th}$. Thus, if we start off with s_j and find segments similar to it, we can end up with a different group of similar segments than if we start off with s_i, even if $d(s_i, s_j) < \delta_{th}$ holds.

Let segment s_i be contained in L_i groups of similar segments, where s_i is similar to $(L_i - 1)$ other segments. The cases where s_i appears as the longest segment (b) and not-the-longest segment (c) are scaled by 2 and 0.1, respectively. For unique (a) segments, we scale by 1.5. These values are empirically chosen. A higher weighting is given for the longest segment (b) and a lower weighting for the shorter segments (c) in order to appropriately emphasize or suppress segments. Case (b) segments are stronger candidates to be included in the summary than unique (a) segments. Let $\mathcal{S}_{i,1}$ (and $\mathcal{S}_{i,0}$) denote the number of times s_i appears as the longest (and not-the-longest) segment among the L_i groups, as in (1) . The weight of all the frames in segment s_i, denoted by $score(i)$, is computed as follows:

$$score(i) = \begin{cases} (\mathcal{S}_{i,1} \times 2 + \mathcal{S}_{i,0} \times 0.1)/L_i & L_i > 1 \\ 1.5 & L_i = 1 \end{cases} \quad (1)$$

These scores for each segment constitute the DTW-based feature. Intuitively, the feature is high for unique segments (a) and the longer length repeated segments (b).

2.3 Feature Fusion

A gradient descent approach is used to derive weights for combining the various feature functions described in Sec. 2.2. Let the values corresponding to k-means, camera, speech, adaptive sampling, and DTW-based features for the n^{th}

frame (after subsampling) be denoted by $f_1[n], \cdots, f_5[n]$, respectively. Thus, $f_{total}[n]$, the "importance" value for the n^{th} frame, is given by:

$$f_{total}[n] = w_0 + \sum_{i=1}^{5} w_i \times f_i[n] \quad (2)$$

A constant w_0 is included to provide a small value for frames that do not take on a specific feature. The f_{total} function is optimally learned by adjusting the weights $\{w_i\}_{i=1}^{5}$ such that it peaks at the ground truth regions.

A gradient descent approach was used to derive feature weights $\{w_i\}_{i=1}^{5}$ based on maximizing the fraction of included events (recall) over a set of videos annotated with the corresponding ground truth. A ground truth event was considered an inclusion if the summarized video contained at least 15 overlapping frames. This descent technique was performed several times starting from different initial weights, as the scheme is vulnerable to local minima.

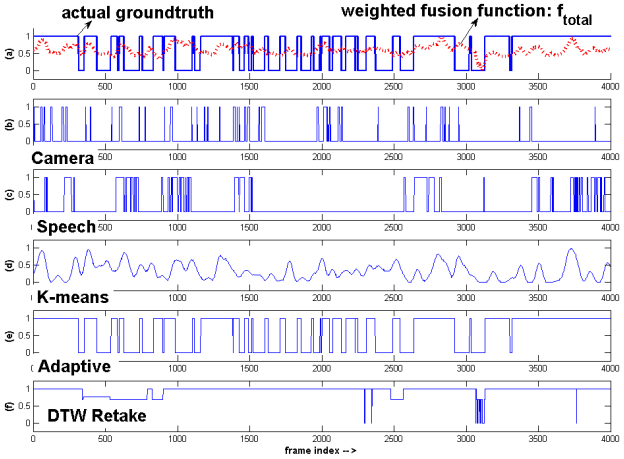

Figure 4: Weighted Fusion of (b)-(f) produces the overall function f_{total} which should correspond with the ground truth, as in (a).

From the f_{total} (2) function the next step is to select candidate keyframes. These keyframes are selected by adjusting the sampling rate proportionately to the area under the weighted importance function. Let the N_c candidate keyframe locations be denoted by $\{F_i\}_{i=1}^{N_c}$ (4). The total "importance" of the video is found by summing over (2) for all the N frames. The video is then divided into sections of equal importance ($\mathcal{I}$) by finding adjacent, nonoverlapping segments with boundaries $\{b_i\}_{i=0}^{N_c}$, with $b_0 = 0$ and $b_{N_c} = N$.

$$\mathcal{I} = \sum_{n=b_{i-1}}^{b_i} f_{total}[n] = \sum_{n=0}^{N} f_{total}[n]/N_c, \ 1 \le i \le N_c \quad (3)$$

The algorithm, beginning with $b_0 = 0$, sums the segment importance until it reaches the value $\mathcal{I}$, as in (3), labeling that location b_1. This process is repeated for all i to define $\{b_i\}_{i=1}^{N_c}$. The candidate keyframes are selected at the midpoints of these equal-importance segments.

$$F_i = \frac{b_{i-1} + b_i}{2} \quad (4)$$

2.4 Postprocessing

The candidate keyframes selected after feature fusion still contain irrelevancies that should be removed. Specifically, postprocessing sought to remove color bars, monochromatic frames, and clapboards, which were particular to the rush videos being summarized. While it would be preferable to remove these irrelevant frames at the start, computational limits precluded their detection on the full video.

2.4.1 Color Bar and Monochromatic Frame Removal

Color bars were found to appear at the start of many videos. Template matching between 1125-dimensional localized color histograms identifies color bars. A color bar frame, with piecewise uniform intensity and without noise, was selected for use as a template. The threshold for the distance metric was determined empirically.

In addition, monochromatic frames are present in between shot changes or at the end of the videos. The frames are identified by measuring the entropy of their color histogram. Since there is low intensity variation within each monochromatic frame, a coarse 20-bin histogram was sufficiently precise for identification. Frames having an entropy value less than an empirical threshold are eliminated.

2.4.2 Irrelevant Object Removal

An attempt to recognize and subsequently remove clapboards, often occurring between shots, is made using a bag-of-features distance metric with scale-invariant feature transforms (SIFT). These descriptors are shown to be relatively robust to changes in size, rotation, and perspective [7]. Specifically, the Caltech-101 image set was used to create a SIFT vocabulary tree that was then used to classify frames using an algorithm based on the work of Nister and Stewenius [11] and derived from an implementation provided by Vedaldi [1]. Each image is characterized by a signature that gives the number of descriptors that go through each node in the vocabulary tree. The distance between a test image and a database image is performed as a weighted L_1 distance between each of these node frequencies.

On unseen development data, testing showed a true positive rate around 90% and a false positive rate less than 2%. False negatives occur because of blurred frames, objects seen at extreme angles, or objects that are dominated by the background. Blurred frames can be addressed by applying a "smoothing" filter where object recognition is performed over a window around the candidate keyframe. Computational limits with the current setup precluded such smoothing, though on certain focus development sets smoothing improved true positive rates by up to 10%.

2.5 Final Clustering

After observing output summary examples it became clear that many still contained a high proportion of redundant shots. Therefore, a final k-means clustering is performed on the candidate keyframes $\{F_i\}_{i=1}^{N_c}$ (4) selected from the function f_{total} (2) after irrelevant objects are removed. The number of clusters is set such that the final summaries are 4% of the total video length, and keyframes closest to the centroids are selected and then padded with 15 frames on each side to build the final summary. The 15 frame padding was chosen because shots less than 1 second in duration have been found to be perceptually frustrating. The rush videos were 25 frames per second. The relatively "most important"

4% of the video is kept in the final summary. The final output summaries did not include any audio.

3. RESULTS

The frames at which the annotated ground truth events occurred for 20 development videos were labeled in order to find the fusion weights that maximized recall. The weights (0.09,0.49,0.55,1.00,1.00,0.55) for (constant,speech,camera,k-means,retake suppression,adaptive sampling) were found using the gradient descent algorithm. By our own scoring method, we found these combined weights to improve recall by 6% compared to a baseline method that created summaries by uniform sampling.

Quantitative results are difficult to create for a subjective task such as evaluating video summarization. Judges viewing the collaborating team summaries indicated that the videos generated by this system performed well as far as inclusion of ground truth key events and were additionally easy to view. Using these weights our summaries were found to include on average over the test videos a fraction of 0.6 of ground truth events, an 8.6% improvement over the CMU baseline uniform summarizations after normalization for average summary length. Table 1 compares our results for the fraction of inclusion metric.

Using NIST scores, the system performance relative to other comparable systems produced summaries that:
1) had an above average inclusion of key events,
2) were easy to interpret,
3) contained an average amount of redundant inclusions, and
4) were longer in duration than other summaries.

The system run-time, in an unoptimized framework, was rather high. For example, a video (MRS025913) of length 25.42 mins took approximately 3.5 hours to summarize when run on a Pentium IV 2.3GHz, 8 GB RAM machine. The biggest slow-downs occurred with feature extraction, DTW distance computation and generation of the high-quality video frames used for the summary.

Table 1: The names of the top 5 teams are listed w.r.t fraction of inclusion of ground truths - both before ($R_{frac}(unnorm)$) and after normalization ($R_{frac}(norm)$) by the summary length. We provide the average inclusion fractions and our own results for comparison.

School	$R_{frac}(unnorm)$	School	$R_{frac}(norm)$
nii	0.6800	hut	0.0173
lip6	0.6700	cityu	0.0156
cityu	0.6400	hkpu	0.0128
cmu	0.6000	thu_icrc	0.0127
ucal	0.6000	eurecom	0.0126
avg.	0.4795	avg.	0.0100
ucal	**0.6000**	**ucal**	**0.0103**

4. ANALYSIS AND CONCLUSIONS

While visual examination and NIST scores reveal a high degree of accuracy as regards inclusion of important video elements, the method described is highly computationally intensive. The greatest slowdown occurs during low-level feature extraction, both for HSV descriptors and SIFT descriptors. The SIFT descriptors are used only for object recognition, and unless such object recognition is highly important the costs outweigh the algorithm's ability to accurately perform recognition. Future work could focus on finding an alternate cutoff point for inclusion in the summary, such as absolute importance or a break in the importance distribution $f_{total}[n]$ (2).

Empirically a big change was seen when a second-pass at k-means was used for clustering. A second pass for redundancy removal provides scope for further improvement. After the candidate keyframes are selected, a similar/duplicate frame detection scheme can be run to further detect repeats in the summary, alleviating reliance on the final k-means clustering step.

5. ACKNOWLEDGMENTS

The group would like to thank Jelena Tesic, IBM T.J. Watson Research Center and Joriz De Guzman in the Vision Research Lab. Support provided by NSF IGERT Grant # DGE-0221713.

6. REFERENCES

[1] http://vision.ucla.edu/∼vedaldi/code/bag/bag.html. UCLA Bag of Features.

[2] P. Bouthemy, M. Gelgon, and F. Ganansia. A unified approach to shot change detection and camera motion characterization. *IEEE Trans. on Circuits and Systems for Video Technology*, 9(7):1030–1044, 1999.

[3] Y. Gong and X. Liu. Video summarization and retrieval using singular value decomposition. *Multimedia Systems*, 9(2):157–168, 2003.

[4] A. Hanjalic. Generic approach to highlights extraction from a sport video. In *Proc. of ICIP (1)*, pages 1–4, 2003.

[5] A. Hanjalic. *Content-Based Analysis of Digital Video*. Kluwer Academic Publishers, 2004.

[6] Y. Li and C.-C. J. Kuo. *Video Content Analysis using Multimodal Information*. Kluwer Academic Publishers, 2003.

[7] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110, 2003.

[8] L. Lu, H. Jiang, and H. Zhang. A robust audio classification and segmentation method. In *ACM Multimedia*, pages 203–211, 2001.

[9] Y.-F. Ma, L. Lu, H.-J. Zhang, and M. Li. A user attention model for video summarization. In *ACM Multimedia*, pages 533–542. ACM Press, 2002.

[10] Y. Nakamura and T. Kanade. Semantic analysis for video contents extraction - spotting by association in news video. In *ACM Multimedia*, pages 393–401, 1997.

[11] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. of CVPR*, pages 2161–2168, Washington, DC, USA, 2006.

[12] P. Over, A. F. Smeaton, and P. Kelly. The TRECVID 2007 BBC rushes summarization evaluation pilot. In *In Proceedings of the TRECVID Workshop on Video Summarization (TVS'07)*, pages 1–15. ACM Press, New York, NY, Sept 2007.

[13] L. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall Signal Processing Series, Englewood Cliffs, New Jersey, 1993.