Hindawi Publishing Corporation
Journal of Biomedicine and Biotechnology
Volume 2012, Article ID 215019, 12 pages
doi:10.1155/2012/215019

Research Article

Using Hierarchical Time Series Clustering Algorithm and Wavelet Classifier for Biometric Voice Classification

Simon Fong

Department of Computer and Information Science, University of Macau, Taipa, Macau

Correspondence should be addressed to Simon Fong, ccfong@umac.mo

Received 22 December 2011; Accepted 25 December 2011

Academic Editor: Sabah Mohammed

Voice biometrics has a long history in biosecurity applications such as verification and identification based on characteristics of the human voice. The other application called voice classification which has its important role in grouping unlabelled voice samples, however, has not been widely studied in research. Lately voice classification is found useful in phone monitoring, classifying speakers' gender, ethnicity and emotion states, and so forth. In this paper, a collection of computational algorithms are proposed to support voice classification; the algorithms are a combination of hierarchical clustering, dynamic time wrap transform, discrete wavelet transform, and decision tree. The proposed algorithms are relatively more transparent and interpretable than the existing ones, though many techniques such as Artificial Neural Networks, Support Vector Machine, and Hidden Markov Model (which inherently function like a black box) have been applied for voice verification and voice identification. Two datasets, one that is generated synthetically and the other one empirically collected from past voice recognition experiment, are used to verify and demonstrate the effectiveness of our proposed voice classification algorithm.

1. Introduction

Every human voice is unique [1] as it was found to be quantitatively composed of components called phonemes that have a pitch, cadence, and inflection. Hence human voice has been used as one of the popular biometrics in biosecurity applications; it can be used to authenticate a person's identity (identification) and control access (authentication and verification) to a protected resource. Unlike other biological traits, like fingerprints and iris scans, voiceprints are relatively vulnerable to replay attack. Much of the research works have been devoted to finding improved solutions in the hope of strengthening voiceprints for meeting high demands of security applications. Some popular techniques include multimodal authentication that fused audio, visual, and other forms of biometrics into one [2]. Since then, voice biometrics has been largely geared towards the security directions of biometric identification and biometric verification. Voice biometrics is used either alone or in combination with other biometrics. In voice verification (VV), a voiceprint of a speaker who claims to be who he is, is presented to the biometrics system for a one-to-one checking of the reference voiceprint which is stored in a database. Once he is successfully verified with a match, subsequent access rights would be granted to him. The other type of checking called voice identification (VI) relies on a one-to-many checking for identifying a previously unknown voiceprint. The unlabelled voiceprint under question is searching through the whole database with the aim of finding a match of an already known sample.

We can see that both VV and VI require a priori condition that a set of voiceprints must have already been known for the matching of new samples to proceed. This is akin to database query or supervised learning where preknown samples must be initially used to train up a decision model, so testing and matching of new sample can follow. A generic example is illustrated in Figure 1. What if in a scenario where a handful of unknown voiceprints are collected, but we wish to obtain some information about them? Such scenarios may include but not limited to security surveillance

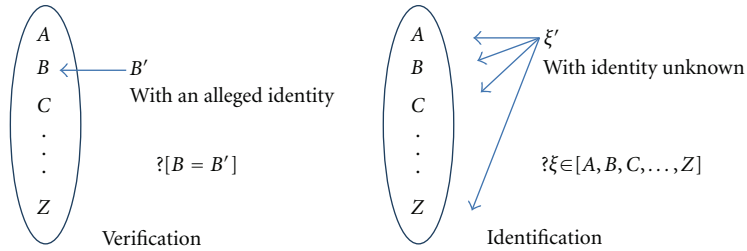


Figure 1: Workings of voice verification and voice identification systems.

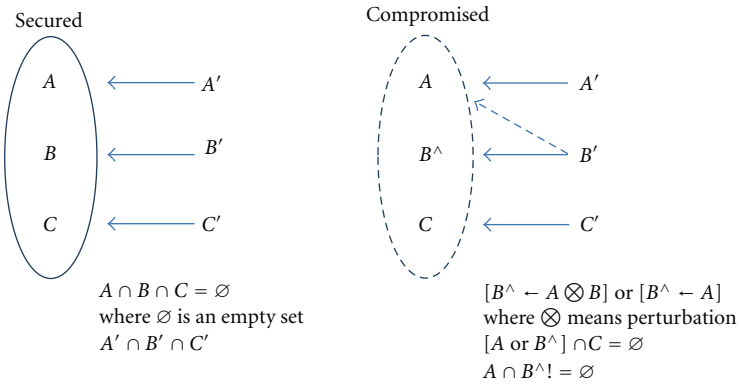


Figure 2: Example that shows a voice-biometric system is compromised, and a voiceprint is counterfeited.

problems [3] where a list of voice traces are captured from a monitored area, how many unique speakers there are, their ages, and genders, and from their speech accents which ethnic backgrounds these people belong to; customer-service applications where callers will be automatically classified from their tones to categories of their needs and emotions. It was only until recently, voice classification (VC) that attempts to determine if a speaker should be classified to a particular characteristic group rather than to a particular individual has gained popularity. VC can help complement the security of VV and VI systems too. In Figure 2 an example of a voice biometric system is being compromised; through hacking, the content of a voiceprint B is modified to that of another voiceprint (let us say A) that has a higher access authority. That can be done by replay attack or injecting vocal features of A into B. Because the database of the voiceprints just like an encrypted list of passwords in a file system is accessed individually, each voiceprint is protected independently; allowing the existence of two same voiceprints goes undetected. So an imposter with B′ can cheat gaining a restricted access right by matching B′ to A in a VI system. VC could be used to prevent this fraud by checking how many unique items there are in different groups. If extra voiceprints suddenly emerge or have gone missing from a group, the integrities of the voiceprints must have changed.

For developing a VC system, several approaches have been studied, such as Artificial Neural Networks (ANN), Support Vector Machines (SVMs), Hidden Markov Models (HMMs) and Gaussian Mixture Models (GMMs). They have been used heavily for training up a model with predefined voice samples for voice recognition. Table 1 shows a summary of the techniques by which majority of research works used. These techniques generally function like a black box; for instance, the weights for mapping the relations of the inputs to the outputs are in plain numeric, the kernel parameters are used for low-level computation, and so forth. They require a full set of known samples to be available before they can be tuned up for actual use. In this paper we propose a fundamentally new approach by using unsupervised learning—clustering, where priori labeled samples are not needed—the characteristic groupings will be dedicated by the samples themselves. Voiceprints who share similar features will be placed into distinctive groups that represent some labels about the speakers. Subsequently a decision tree (classifier) can be built after studying and confirming the characteristic groups. The classifier will then be used for classifying new samples into the groups. The advantage of decision tree is that easily comprehensible rules in terms of IF-THEN-ELSE conditions can be generated when the decision tree is constructed. That gives an edge over the aforementioned black-box types of classification algorithms. Using the features of a human voice as a voice classifier for classifying speakers has not been researched a great extent although it has a large implication in voice biometrics applications. To the best of the author's knowledge, nobody has applied such techniques of VC before. This is the research focus of this paper; a collection of algorithms are introduced for supporting grouping unlabeled voiceprints and then subsequently classifying new incoming voiceprints. They can be used for checking the integrity of the groups of voiceprints for solving

Table 1: Classification algorithms where majority of research works on voice classification used.

	ANN	HMM	GMM	SVM
Kanak et al. [8]		✓		
Nefian et al. [9]		✓		
Fox et al. [10, 11]		✓		
Bengio [12, 13]		✓		
Chaudhari et al. [14]			✓	
Aleksic and katsaggelos [15]		✓		
Wark et al. [16–18]			✓	
Jourlin et al. [19]		✓		
Hazen et al. [20]				✓
Sanderson and Paliwal [21]			✓	
Ben-Yacoub et al. [22]		✓		
Chibelushi et al. [23]	✓			
Luettin et al. [24]		✓	✓	
Moreno and Ho [25]				✓

the security problem that is illustrated in Figure 2. The contribution of this paper is an alternative computation platform for realizing voice classification; the algorithms are relatively simpler than the existing ones and fellow researchers that can easily adopt them for implementing VC systems.

2. Our Proposed Model

The model that we proposed aims at providing a generic voice classification framework under which a collection of algorithms such as hierarchical time series clustering, dynamic time wrap transform, discrete wavelet transform and decision tree would have to work together. The prominent advantage is its generic property that can be applied across a variety of applications that capitalize on voice classification. While the inputs are previously unknown voices, the voices would be automatically grouped together according to their own characteristics. Each group or cluster being formed as an output represents one pronounced characteristic which is shared in common by all the voice samples inside (total = n). Our model assumes that the collected voices in waveforms would be recorded in the format of time series. Each time series is a vector of numeric data points that can be represented by a set of m attribute values, such as a time series $s = [x_1, x_2, \ldots, x_m]$. In a collected dataset whose speakers' identities are not known, a sufficient amount of voice samples are gathered from each speaker and these samples can then be clustered by using hierarchical time series clustering algorithm. Clustering is done based on the characteristics of the voice samples themselves. At this point no classification area was sought specifically, for instance gender or ethnic background, as it was preferred to allow the results to decide the characteristics that lead to a particular clustering group.

As shown in Figure 3, an example scenario by the proposed model is a surveillance eavesdropper that collects from a secret meeting a total of n voice traces. The voice traces may be spoken by more than one speaker, one trace per speaker at a time, and each voice trace can be encoded by m coefficient attributes regardless of how long the conversation is. The voices are assumed to be undistorted and not intermixed. The voices that are in the form of time series can be submitted for hierarchical clustering for self-grouping. Hierarchical clustering instead of others is applied because it gives a layered structure of groupings which we do not know in advance in different resolutions. After the clustering, not only we know how the speakers whose voices are distinctively grouped, the number of unique voices (hence the number of speakers) can also be identified. In essence, it may be possible to infer from the groupings that how many speakers there are in the meeting, what characteristics they have in each group. However, it requires further verification and probably extra information to infer detailed assertions such as gender, age, and the emotions of the speech.

With the groupings available, the voice analysts can assign meaningful labels on the groups. A voice classifier can be developed after the unlabeled voice traces labeled with the classes derived from the characteristics of the groups. So that in our model, unsupervised learning by clustering comes first, and then supervised learning for building the decision tree follows. The voice traces in the form of labeled time series can be used as training data to build a classifier. However, in our experiment, we opt to transform the voice traces from time domain to frequency domain, as our experiment results show that the accuracy performance of the classifier can be significantly improved. When the classifier is ready, future new voice samples can be automatically classified into the characteristic groups. If necessary, the process of hierarchical clustering can be applied on the new samples again in case new characteristics from the voice samples may be discovered.

3. Design of the Hierarchical Time Series Clustering Algorithm

The goal of time series clustering is to identify the speaker category to which a voice belongs given the multivariate time series points of each voice trace. In our experiments the synthetic control wave dataset and empirical datasets from UCI data archive were used [4]. The wave has 60 coefficients and the live Japanese vowel data each wave is characterized by 12 coefficients. The time series data are grouped together based on similarity—similar waves cling together to form a cluster, and dissimilar waves tend to stay far apart in separate clusters. Iteratively the time series clustering algorithm relocate the data points one step at a time to ensure that the data points inside the same cluster have the minimum intradissimilarity and data points across different clusters have the maximum inter-similarity. The similarity is defined as the multidimensional distance between two data points whose multiple attributes are measured as how close they are in values. Two variables exist

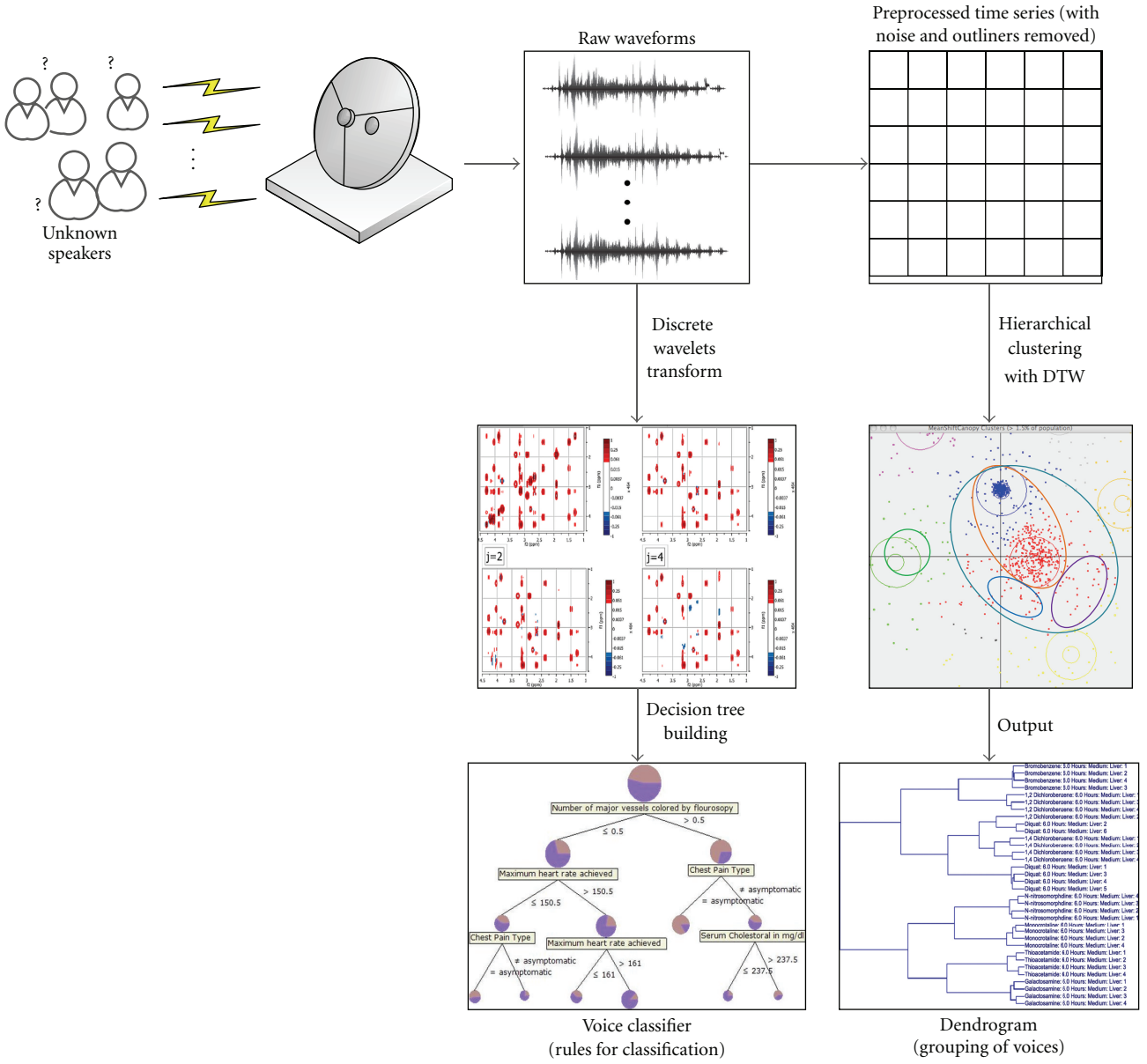


Figure 3: The proposed model of voice classification with hierarchical clustering.

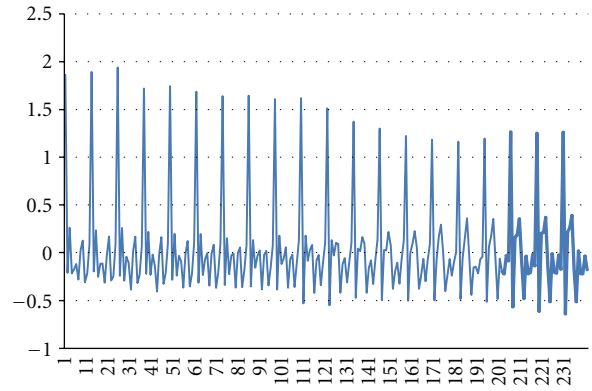


Figure 4: A sample time series represented in LPC coefficient produced by utterance of Japanese vowels.

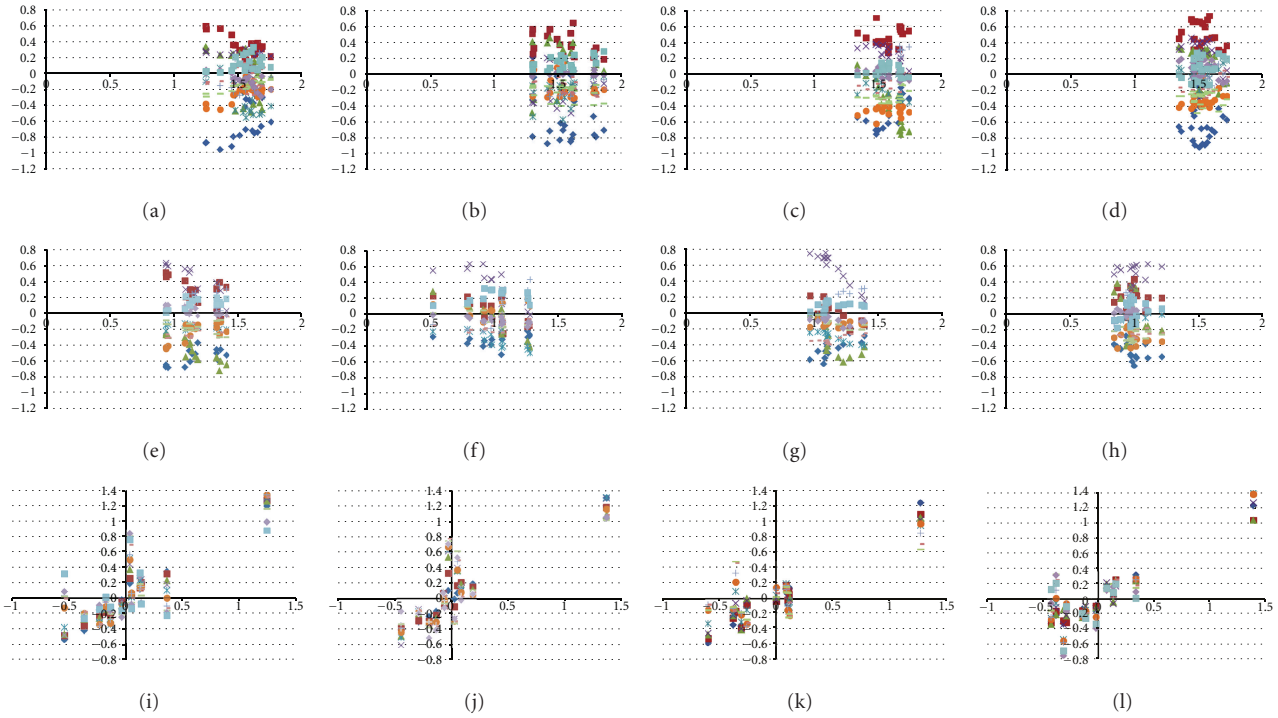


Figure 5: Visualization of time series plots that represent the voiceprints by three different speakers who uttered the same Japanese vowels.

for time series clustering algorithm, one is for choosing the similarity function for measuring the distance between each pair of data points, and the other is the overall operation that converge from an initial assignment of data points to clusters to a converged or optimal assignment of data points to clusters.

Many similarity measures are available such as Manhattan, Euclidean and Minkowski just to name a few. In our experiments, a range of popular similarity functions are compared in performance in order to observe which one performs the best. Table 2 shows a list of performance results in the percentage of correctly clustered groups by using various similarity functions. Because the nature of the data points that we are working with is time series, we choose to use Dynamic Time Warping function (DTW) as a distance measure that finds optimal alignment between two sequences of time series data points. DTW a pairwise comparison of the feature (or attribute) vectors in each time series. It finds an optimal match between two sequences that allows for stretched or compressed sections of the sequences. In other words it allows some flexibility for matching two sequences that may vary slightly in speed or time. The sequences are "warped" nonlinearly in the time dimension to determine a measure of their similarity independent of certain nonlinear variations in the time dimension. It is popular in the application of signal processing where two signal patterns are to be matched in similarity. Particularly suitable DTW is for matching sequences that may have missing information or various lengths, on condition that the sequences are long enough for matching. In theory, DTW is most suitable for voice wave patterns because exact matching for such patterns often may not occur, and voice wave patterns may vary slightly in time domain. A comparison will be given in our experiment to verify this hypothesis. The pseudo code of the DTW algorithm is given in Algorithm 1.

For clustering time series, likewise many variants of algorithm are applicable. They range from simple ones like K-means and K-medoids, to sophisticated algorithms like DBSCAN, density-based clustering for clustering structures. In our case, hierarchical clustering is desirable because it allows the time series which are voice waves to be grouped in different levels automatically that helps a user to explore the structure of the groupings from coarse to refined. This is particularly useful when the grouping structure is not known in advance. Like most of the clustering algorithms which operate by unsupervised learning, hierarchical clustering does not require the number of clusters to be predefined at the beginning; it allows the data to decide the suitable number of groups by themselves. In our experiment, agglomerative mode which is also known as the "bottom up" approach is used. Initially each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy. For deciding which clusters should be merged, a similarity function is used between sets of observations. A variety of similarity functions are used here for experiments, they are Canberra, DTW, Euclidean, Manhattan, and Minkowski from power 1 to 10. The clustering algorithm constructs the hierarchy from the individual time series by progressively merging clusters up. The basic process of hierarchical clustering comprises of the following steps, given n time series, and a two dimensional $n \times n$ similarity matrix S.

```
DTW(v1, v2)  {
//where the vectors v1=(a1,...,an), v2=(b1,...,bm) are the time series with n and m
time points
    Let a two dimensional data matrix S be the store of similarity measures
such that S[0,...,n, 0,...,m], and i, j, are loop index, cost is an integer.
    // initialize the data matrix
    S[0, 0]:= 0
    FOR i := 1 to m DO LOOP
        S[0, i]:= ∞
    END
    FOR i := 1 to n DO LOOP
            S[i, 0]:= ∞
    END
    // Using pairwise method, incrementally fill in the similarity matrix
    with the differences of the two time series
    FOR i := 1 to n DO LOOP
        FOR j := 1 to m DO LOOP
        // function to measure the distance between the two points
            cost := d(v1[i], v2[j])
            S[i, j] := cost + MIN(S[i-1, j],           // increment
                                  S[i, j−1],    // decrement
                                  S[i-1, j−1])  // match
        END
    END
    Return S[n, m]
}
```

Algorithm 1: Pseudo code of dynamic time wrap algorithm.

Table 2: Percentage of correctly clustered groups in various similarity functions.

Canberra	DTW	Euclidean	Manhattan	Minkowski	Minkow. 2	Minkow. 3
86.67	91.67	63.33	63.33	63.33	63.33	66.67
Minkow. 4	Minkow. 5	Minkow. 6	Minkow. 7	Minkow. 8	Minkow. 9	Minkow. 10
78.33	83.33	86.67	76.67	66.67	66.67	66.67

Step 1. Each time series is assigned to a cluster of its own, with a total of n clusters for n time series. Initialize S with similarity measures between the clusters which are the same as the similarity measure between the time series that they contain.

Step 2. The most similar pair of clusters are merged into a single cluster. Retain the current level of clusters and move up a level in the hierarchy.

Step 3. Calculate the new similarity measures in S between the new clusters and each of the old clusters.

Step 4. Finally repeat Step 2 and Step 3 until all the time series are clustered into a single cluster of size n. When this happens, the highest level of the hierarchy is attained.

4. Experiments

4.1. Datasets. Two experiments are conducted for testing the performance of the algorithm over a synthetic control dataset [5] and a live dataset [6]. The synthetic control dataset contains 600 examples of time series wave forms synthetically generated by the process in Alcock and Manolopoulos (1999) [7]. There are six different classes of control charts that represent generally different shapes of time series waves. Each wave is characterized by 60 coefficients in the form of temporal data points, each different group (or class) has 100 samples and total there are 600 samples in the dataset. Each class has an essentially unique bunch of waveforms that are different from those of the other classes; hence, we can assume the six classes represent six different types of speakers who have different voiceprints from one another. For example, it could be speakers who come from six different geographical locations therefore different ascents, speakers who speak in six different emotions/languages, or speakers of six distinctively different age groups. We generalize the names of these characteristic groups as speech types.

The other empirical dataset contains 640 time series of 12 linear prediction cepstrum coefficients (LPCs) taken from nine male speakers. The data was initially collected by a Japanese research team for examining a newly developed classifier for multidimensional curves (multidimensional time series). The volunteered speakers uttered two Japanese

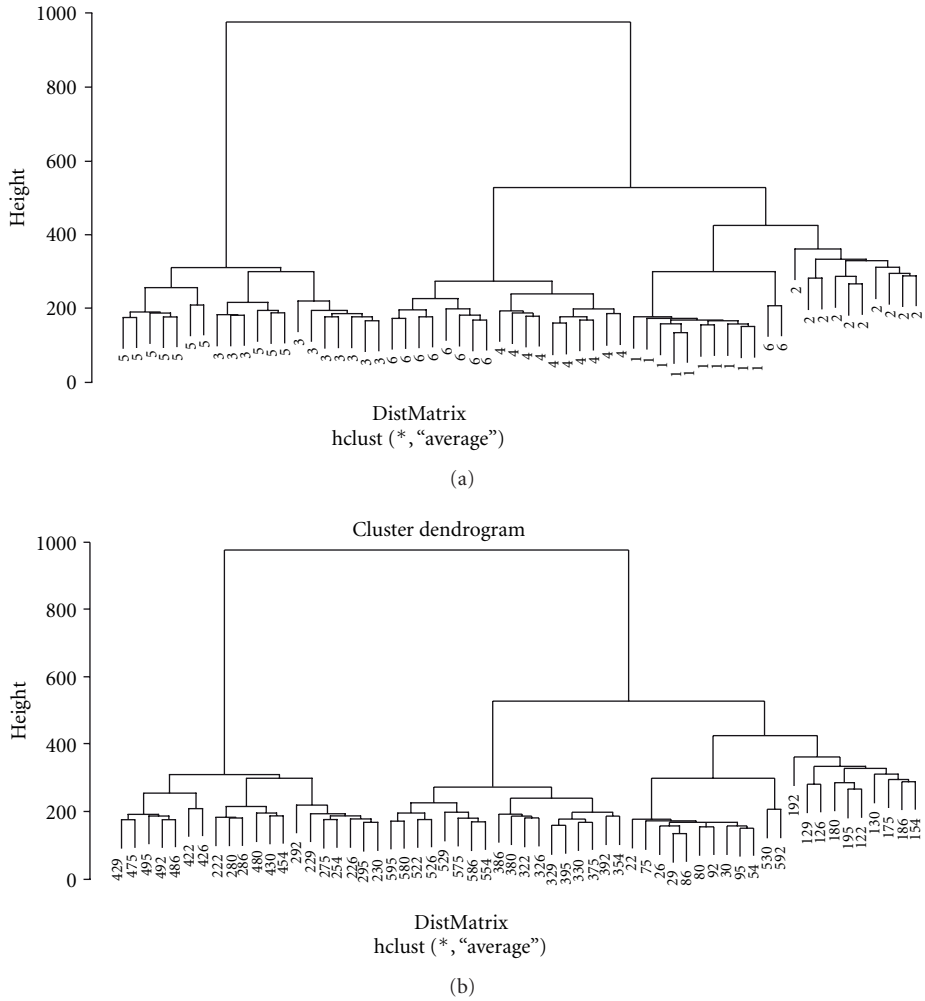


Figure 6: (a) Six characteristic groups at the dendrogram by using DTW similarity function. (b) The corresponding row numbers of the dataset at the dendrogram by using DTW similarity function.

vowels /ae /successively. For each utterance, a 12-degree linear prediction analysis was applied to obtain a discrete-time series with 12 LPC cepstrum coefficients. This means that one utterance by a speaker forms a time series whose length is in the range 7–29, and each point of a time series is of 12 features (12 coefficients). Analysis parameters are as follows: sampling rate = 10 kHz, frame length = 25.6 ms, and shift length = 6.4 ms. So for the dataset, a set of consecutive blocks represents a unique speaker. There are 30 blocks for each speaker. Blocks 1–30 represent speaker 1, blocks 31–60 represent speaker 2, and so on up to speaker 9. A sample of the time series taken from one of the voice trace is shown in Figure 4.

Just as shown in our model in Figure 3, the raw voice series are formatted and processed into records that have exactly 12 coefficients (attributes). Hierarchical time series clustering is applied over the processed data, so that each data point that the clustering algorithm works with has identical attributes and scales for similarities measures. By plotting the processed data with x-axis as the first column of a consecutive block against the rest of the series with values within the range at the y-axis, we generate some visualization of the time series points with distinguishable shapes. Figure 5 shows three groups of voice series that are taken from the dataset blocks from three different speakers. Just by visual inspection, we could observe their differences in appearance. The four voice utterances on the top row sit at about three quarters on the x-axis, the cap of the data clusters is dominated by small square dots (that just represent one of the coefficient values of the block of the sample), then followed by other shapes of dots and diamond shaped dots at the bottom. Though each of the four clusters on the top row is not exactly identical to each other, they roughly have a similar structure. In the middle row, the voice visualization by another speaker has the data near the middle of x-axis, and the outlined structure has the cross-shaped markers on the cap. And the visualization on the bottom row has an obviously different formation than the other two. That shows the voices of the three speakers are essentially different as by their voice characteristics, and the differences can be visually

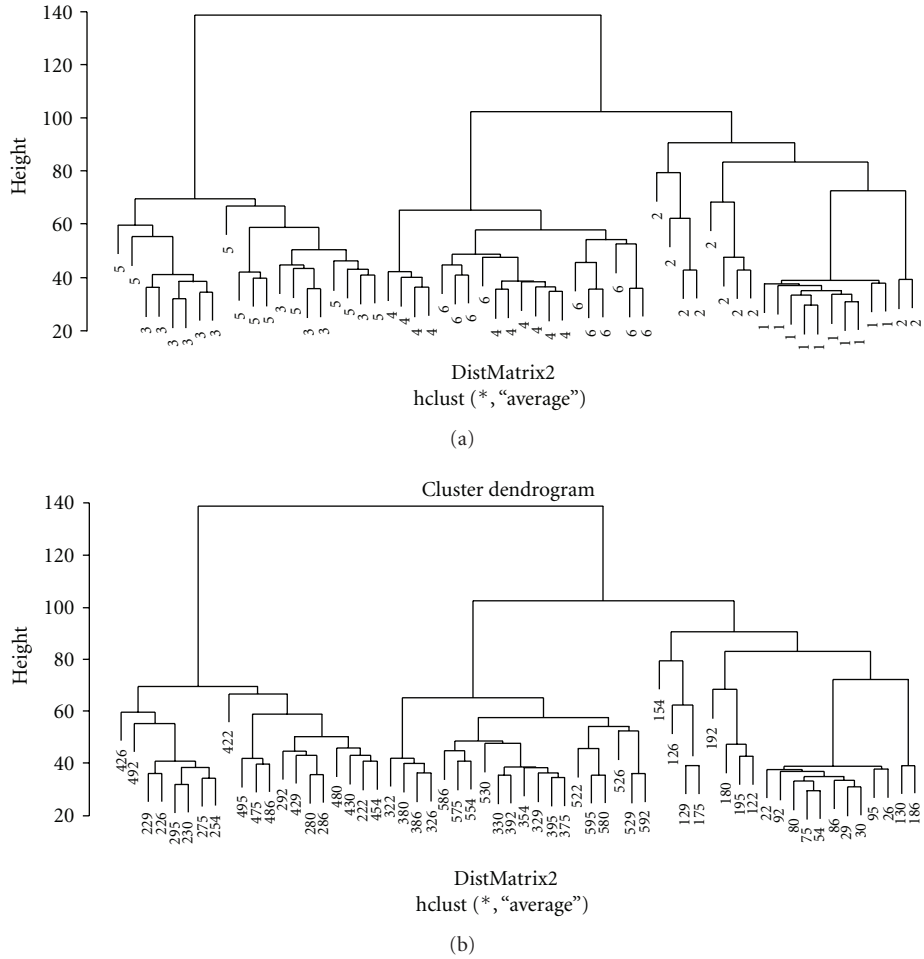


Figure 7: (a) Six characteristic groups at the dendrogram by using Euclidean similarity function. (b) The corresponding row numbers of the dataset at the dendrogram by using Euclidean similarity function.

spotted. However, computationally, the differences in voice characteristics would have to be revealed by clustering algorithm.

4.2. Results and Comparison. The hierarchical clustering algorithm that we applied in the experiment is implemented in R which is a free software environment for statistical computing and graphics (http://www.r-project.org/). The synthetic data are first sampled with a ratio of 10% for producing the first iteration of data points and clusters. DTW that serves as the similarity function is embedded in the clustering algorithm for processing the time series data till convergence. The experiment is repeated with other similarity functions for comparison. A snapshot of the resulting dendrogram by using DTW similarity function is shown in Figure 6(a). It can be seen that the dendrogram by DTW can effectively partition the time series into six distinct groups that represent six speech types. The groupings, by the DTW dendrogram as shown in Figure 6(b), can be used to map over to the actual row number of the dataset that has a total of 600 rows. In other words, the time series that are indexed by the row numbers can be allocated to the six groups by the dendrogram as a result of time series clustering. In another counter example, similarity function by Euclidean distance is applied in the experiment; we can easily see that the groupings at the dendrogram as shown in Figure 7(a) are not in perfect order at all. We can safely remark that DTW is superior to Euclidean in the clustering.

From a bird-eye view, Figure 8 shows clearly on the efficacy of the two similarity functions in hierarchical time series clustering. DTW has generally less chaotic grouping than Euclidean similarity function. Without showing the dendrogram for each of the other varieties of similarity functions, a comparison table below shows the performance of grouping of each technique. The performance is estimated by counting the number of mislocated groups in the dendrogram by that particular similarity function. DTW performs consistently well, while Minkowski shows an optima when the power was increasing from 1 to 10. This observation confirms that DTW is suitable for time series clustering as time series do vary in time domain more or less to certain extent.

The subsequent experiment is to build a decision tree after the grouping has been formed by hierarchical time

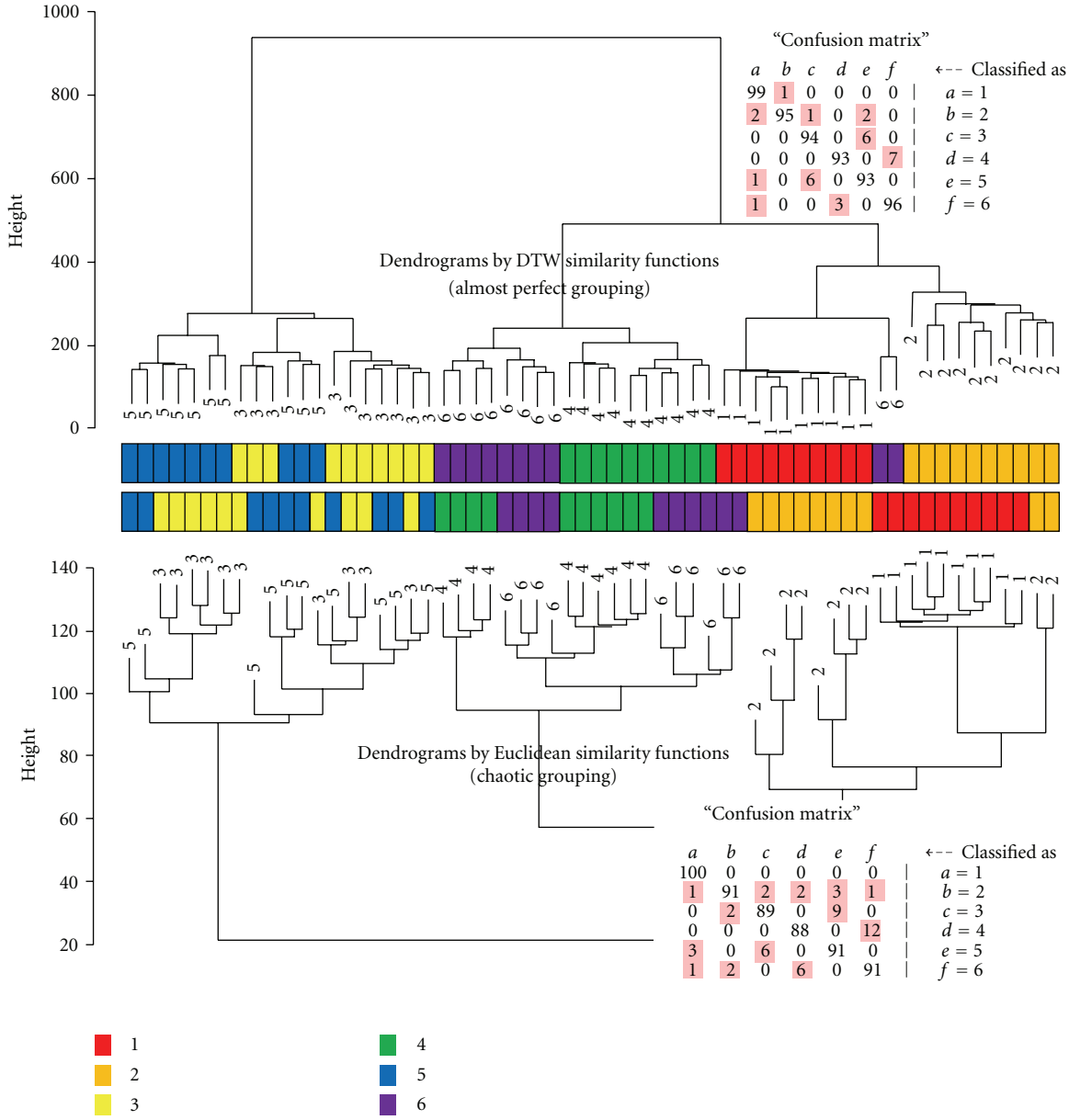
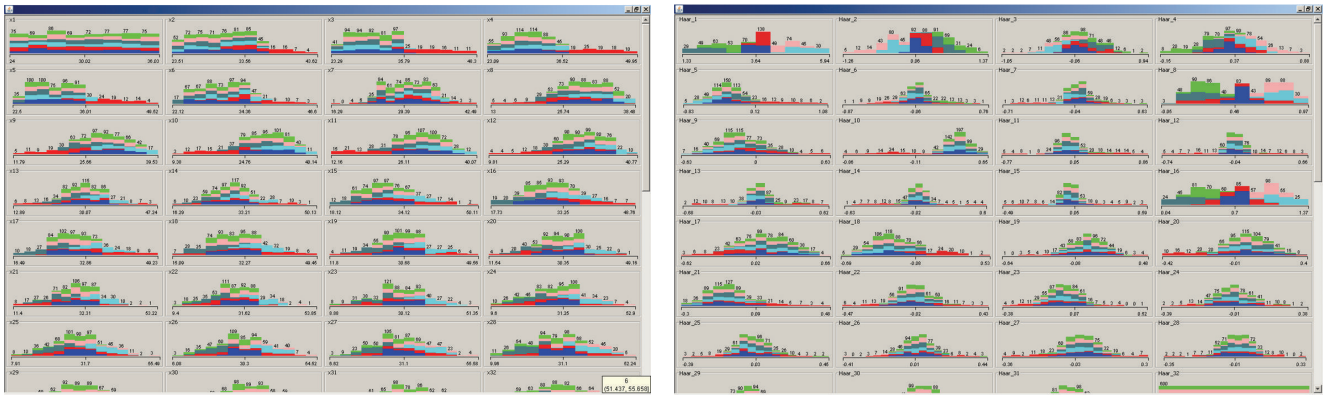


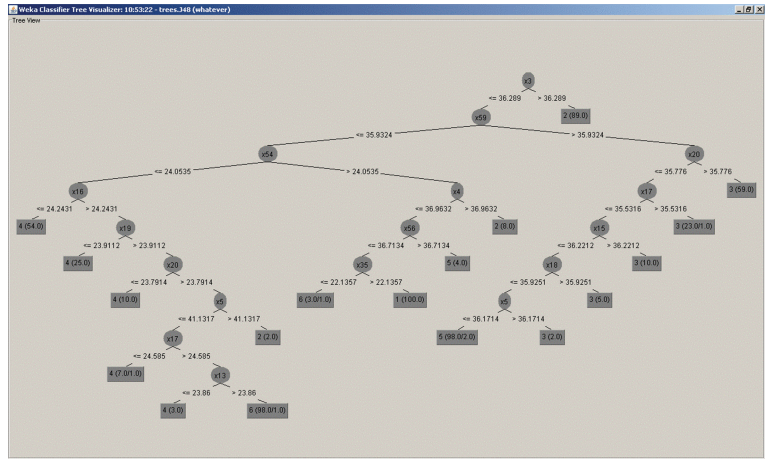
Figure 8: A bird-eye view of the comparison of the dendrograms which are produced by DTW and Euclidean similarity functions respectively.

series clustering. There are two choices of decision trees to be recommended. RIPPER function is suggested to be run for generated comprehensible rules that are in the form of IF-THEN-ELSE. The rules specify a sequence of conditions meeting which in order lead to a predefined class label. When a new voiceprint is received, pass it over the rules by checking its coefficient values that can determine which class label the voiceprint fits in. The other decision tree algorithm is the classical C5.0 or J48 with pruning mode on, in WEKA which is an open source of machine learning algorithms for solving data mining problems implemented in Java and open sourced under the GPL (http://archive.ics.uci.edu/ml). The time series data, however, are converted to their corresponding frequency domain by Discrete Wavelet Transformation (DWT). DWT applies the the Haar wavelet transform which was invented by Kristian Sandberg from University of Colorado at Boulder, USA in year 2000. DWT in principle works better than time series points in classification because DWT can find where the energies are concentrated in the frequency domain, and remarkable coefficients called Haar attributes are well describing the characteristics of the time series. A comparison of the original coefficients in time domain and transformed coefficient in frequency domain can be seen that wavelets after the transformation have sharper and narrower statistical distribution than the time series points, in Figure 9. DWT is implemented in the plug-in filter in WEKA called "weka.filters.unsupervised.attribute.Wavelet."

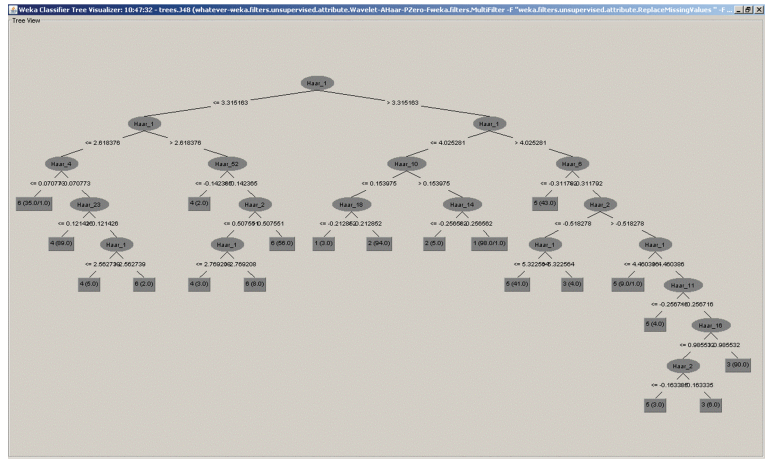


(a) (b)

Figure 9: (a) Attributes of the a voice time series; (b) transformed attributes called Haar coefficient of the wavelet representation of the time series.



(a)



(b)

Figure 10: Snapshots of a decision tree as a result of building a classifier by (a) using the original time series, and (b) using the transformed wavelets.

Table 3: Classification accuracy by the two datasets.

	Synthetic data		Empirical data	
	Time series	Wavelets	Time series	Wavelets
% correctly classified instances	91.67	95.00	55.20	64.13
Root relative sq. error	43.86	34.14	72.82	47.80
Coverage of cases	92.33	95.67	81.54	62.28
Precision	0.958	0.99	0.5770	0.6445
Recall	0.910	0.95	0.5481	0.6185
F-score	0.933	0.969	0.5481	0.6308

The performance of the decision tree which is a voice classifier is defined as a composite of accuracy measures. They generally come in the following indices in data mining as (1) the percentage of correctly classified instances, (2) the root relative squared error, (3) coverage of cases, (4) overall Precision; in a classification task, the precision for a class is the number of true positives (i.e., the number of items correctly labeled as belonging to the positive class) divided by the total number of elements labeled as belonging to the positive class (i.e., the sum of true positives and false positives, which are items incorrectly labeled as belonging to the class), (5) overall Recall which is defined as the number of true positives divided by the total number of elements that actually belong to the positive class (i.e., the sum of true positives and false negatives, which are items which were not labeled as belonging to the positive class but should have been), and (6) F-score, which is a measure of a test's accuracy by considering both the precision and the recall of the test to compute the score.

The performance comparison table is given in Table 3. It compares mainly the classification accuracy by using a J48 decision tree in WEKA of the time series version and the Wavelet version of the two testing datasets. It can be noticed that in general Wavelets have improvement over the time series in terms of classification accuracy. The results of the empirical data are generally lower in accuracy than the synthetic control data probably due to its complex and less uniform in the time series structures, plus the normalization effect for limiting the time series into fixed length from its original variable length. However, wavelet transformation still shows its advantage in applying to the empirical data. A sample of the decision tree generated from the experiment is shown in Figure 10. By using the decision tree as classifier, new voiceprint can fit into a specific class by traversing the decision tree.

5. Conclusion

Using voice as a biometrics has its advantage because it is a noninvasive nature process in human interaction, and human voice has been proven to contain biological traits that can uniquely identify an individual. In the past many studies have focused on applying voice biometrics into security-related applications such as user verification and biometric identification. In contrast voice classification has not been researched extensively. Voice classification is recently becoming popular as it serves as the underlying technique for monitoring different types of speakers and providing supreme customer service by estimating the natures of phone/Web calls; these applications potentially have high values in security surveillance and commercial uses. In this paper, a set of relatively simple and transparent techniques are described for enabling voice classification. Fellow researchers are encouraged to test out the collection of algorithms as recommended in this paper for experimenting voice datasets pertaining to voice classification applications. In particular, we showed via experiments that hierarchical time series clustering algorithm with various similarity functions can yield different levels of accuracy. It is shown possible that time series can be grouped into different clusters, just as if some unknown voices are grouped together by their common characteristics. Wavelets after transforming of time series samples into frequency domain demonstrate an improved accuracy performance in decision tree. The future works include fine-tuning the mentioned algorithms in the paper for even better performance. The algorithms should be programmed into a single software program in order to support as a core classification engine for voice biometric application systems.

References

[1] P. Naresh, S.-H. Cha, and C. C. Tappert, "Establishing the uniqueness of the human voice for security applications," in *Proceedings of the Student/Faculty Research Day (CSIS '04)*, pp. 8.1–8.6, Pace University, May 2004.

[2] P. S. Aleksic and A. K. Katsaggelos, "Audio-visual biometrics," *Proceedings of the IEEE*, vol. 94, no. 11, pp. 2025–2044, 2006.

[3] J. Markowitz, "The many roles of speaker classification in speaker verification and identification," in *Speaker Classification I: Fundamentals, Features, and Methods*, C. Mller, Ed., Lecture Notes in Computer Science, pp. 218–225, Springer, 2007.

[4] A. Frank and A. Asuncion, "UCI Machine Learning Repository," Irvine, Calif, USA, University of California, School of Information and Computer Science, http://archive.ics.uci.edu/ml/.

[5] D. T. Pham and A. B. Chan, "Control chart pattern recognition using a new type of self-organizing neural network," *Proceedings of the Institution of Mechanical Engineers*, vol. 212, no. 1, pp. 115–127, 1998.

[6] M. Kudo, J. Toyama, and M. Shimbo, "Multidimensional curve classification using passing-through regions," *Pattern Recognition Letters*, vol. 20, no. 11–13, pp. 1103–1111, 1999.

[7] R. J. Alcock and Y. Manolopoulos, "Time-series similarity queries employing a feature-based approach," in *Proceedings of the 7th Hellenic Conference on Informatics*, Ioannina, Greece, August 1999.

[8] A. Kanak, E. Erzin, Y. Yemez, and A. M. Tekalp, "Joint audio-video processing for biometric speaker identification," in *Proceedings of the IEEE International Conference on Accoustics, Speech, and Signal Processing*, pp. 561–564, Hong Kong, China, April 2003.

[9] A. V. Nefian, L. H. Liang, T. Fu, and X. X. Liu, "A Bayesian approach to audio-visual speaker identification," in *Proceedings of the 4th International Conference Audio- and Video-Based*

Biometric Person Authentication, pp. 761–769, Guildford, UK, 2003.

[10] N. A. Fox, R. Gross, P. de Chazal, J. F. Cohn, and R. B. Reilly, "Person identification using automatic integration of speech, lip and face experts," in *Proceedings of the ACM SIGMM Multimedia Biometrics Methods and Applications Workshop (WBMA '03)*, pp. 25–32, Berkeley, Calif, USA, 2003.

[11] N. A. Fox and R. B. Reilly, "Audio-visual speaker identification based on the use of dynamic audio and visual features," in *Proceedings of the 4th International Conference Audio- and Video-Based Biometric Person Authentication*, pp. 743–751, Guildford, UK, 2003.

[12] S. Bengio, "Multimodal authentication using asynchronous HMMs," in *Proceedings of the 4th International Conference Audio- and Video-Based Biometric Person Authentication*, pp. 770–777, Guildford, UK, 2003.

[13] S. Bengio, "Multimodal speech processing using asynchronous hidden Markov models," *Information Fusion*, vol. 5, no. 2, pp. 81–89, 2004.

[14] U. V. Chaudhari, G. N. Ramaswamy, G. Potamianos, and C. Neti, "Information fusion and decision cascading for audio-visual speaker recognition based on time-varying stream reliability prediction," in *Proceedings of the International Conference on Multimedia & Expo*, pp. 9–12, Baltimore, Md, USA, July 2003.

[15] P. S. Aleksic and A. K. Katsaggelos, "An audio-visual person identification and verification system using FAPs as visual features," in *Proceedings of the Works Multimedia User Authentication*, pp. 80–84, Santa Barbara, Calif, USA, 2003.

[16] T. Wark, S. Sridharan, and V. Chandran, "Robust speaker verification via fusion of speech and lip modalities," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '99)*, pp. 3061–3064, Phoenix, Ariz, USA, March 1999.

[17] T. Wark, S. Sridharan, and V. Chandran, "Robust speaker verification via asynchronous fusion of speech and lip information," in *Proceedings of the 2th International Conference Audio- and Video-Based Biometric Person Authentication*, pp. 37–42, Washington, DC, USA, 1999.

[18] T. Wark, S. Sridharan, and V. Chandran, "Use of temporal speech and lip information for multi-modal speaker identification via multi-stream HMM's," in *Proceedings of the IEEE Interntional Conference on Acoustics, Speech, and Signal Processing*, pp. 2389–2392, Istanbul, Turkey, June 2000.

[19] P. Jourlin, J. Luettin, D. Genoud, and H. Wassner, "Integrating acoustic and labial information for speaker identification and verification," in *Proceedings of the 5th EUR Conference Speech Communication Technology*, pp. 1603–1606, Rhodes, Greece, 1997.

[20] T. J. Hazen, E. Weinstein, R. Kabir, A. Park, and B. Heisele, "Multi-modal face and speaker identification on a handheld device," in *Proceedings of the Workshop on Multimodal User Authentication*, pp. 113–120, Santa Barbara, Calif, USA, 2003.

[21] C. Sanderson and K. K. Paliwal, "Identity verification using speech and face information," *Digital Signal Processing*, vol. 14, no. 5, pp. 449–480, 2004.

[22] S. Ben-Yacoub, Y. Abdeljaoued, and E. Mayoraz, "Fusion of face and speech data for person identity verification," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1065–1074, 1999.

[23] C. C. Chibelushi, F. Deravi, and J. S. Mason, "Voice and facial image integration for speaker recognition," in *Proceedings of the IEEE International Symposium on Multimedia Technologies and Future Applications*, Southampton, UK, 1993.

[24] J. Luettin, N. Thacker, and S. Beet, "Speaker identification by lipreading," in *Proceedings of the International Conference on Spoken Language Processing (ICSLP '96)*, pp. 62–65, October 1996.

[25] P. Moreno and P. Ho, "SVM kernel adaptation in speaker classification and verification," in *Proceedings of the INTERSPEECH 2004-ICSLP*, pp. 1413–1416, INTERSPEECH 2004-ICSLP, Jeju Island, Korea, 2004.