# SVM-BASED SEPARATION OF UNVOICED-VOICED SPEECH IN COCHANNEL CONDITIONS

*Ke Hu and DeLiang Wang*

Department of Computer Science and Engineering
& Center for Cognitive Science
The Ohio State University
Columbus, OH 43210-1277, USA
{huk, dwang}@cse.ohio-state.edu

## ABSTRACT

Unvoiced-voiced portions of cochannel speech contain considerable amounts of both voiced and unvoiced speech and play a significant role in separation. Motivated by recent developments in separation of speech from nonspeech noise, we propose a classification-based approach for unvoiced-voiced speech separation. A new feature set consisting of pitch-based features and gammatone frequency cepstral coefficients is proposed to represent the characteristics of a time-frequency unit. The cepstral features do not rely on pitch and are thus more robust than the pitch-based features to pitch estimation errors. Speaker-independent support vector machines are trained for classification. Results based on the TIMIT corpus show that the proposed algorithm significantly improves unvoiced speech segregation compared to a recent algorithm.

***Index Terms***— Cochannel speech separation, unvoiced speech, voiced speech, unit-level features, classification

## 1. INTRODUCTION

Cochannel speech refers to a mixture of speech signals from two speakers. In cochannel conditions, two talkers are usually not aware of each other and their speech often has a large amount of overlap. Since speech contains both voiced and unvoiced parts, a cochannel speech signal has three different portions, i.e., voiced-voiced, unvoiced-voiced (UV), and unvoiced-unvoiced portions. Unvoiced speech constitutes about 20 to 25% of spoken speech in terms of time durations [1]. Thus, in cochannel conditions, the UV portions cover about 37.5% of all frames. For each speaker, significant amounts of both voiced and unvoiced speech (about 75% of unvoiced frames and 25% of voiced frames) are included in the UV portions.

Model-based methods perform separation by capitalizing on speaker information. They often estimate sources jointly based on models such as hidden Markov models (HMM), Gaussian mixture models or nonnegative matrix factorization (e.g., [2], [3], and [4]). Model-based methods deal with a whole speech signal including the UV portions, but they often require availability of pretrained speaker models and sometimes speaker identities.

We aim to separate unvoiced speech from voiced speech in a speaker-independent way. In computational auditory scene analysis (CASA), feature-based methods have been proposed to separate unvoiced speech. Onsets and offsets are utilized to segment speech [1]. However, they do not differentiate unvoiced and voiced speech. In [5], spectral subtraction is incorporated in CASA to segregate unvoiced speech. The stationarity assumption of noise is relaxed in [5] compared to traditional speech enhancement methods but is still hard to meet in a cochannel scenario. A tandem algorithm in [6] utilizes 6-dimensional pitch-based features (6F) to group voiced speech. The pitch provides a cue to differentiate voiced and unvoiced speech but the separation performance is closely related to pitch estimation accuracy.

From a different perspective, speech separation can be formulated as classification by estimating the ideal binary mask (IBM) [7]. In the IBM, 1 indicates a target dominant time-frequency (T-F) unit and 0 an interference dominant one. One of the first classification-based methods appears in binaural speech separation. In monaural conditions, supervised learning based on the 6F features [8] or amplitude modulation spectrum (AMS) features [9] has also proven to be effective. Recently, a system in [10] combines the 6F features and AMS features and utilizes support vector machines (SVM) for classification. This system improves the segregation performance and also demonstrates good generalization ability to unseen noise.

Inspired by the aforementioned approaches, we propose a classification based method to separate unvoiced-voiced speech in cochannel conditions. Our work differs from those in [9] and [10], which only deal with separation of speech and nonspeech noise. In this work, we explore different features such as the pitch-based features, gammatone frequency cepstral coefficients (GFCC) [11] and the AMS features for classification of T-F units. Results show that GFCCs are competitive features, and we combine them with the pitch-based features and construct a new feature set for classification. Since extraction of GFCCs does not depend on pitch, it can also deal with frames where pitch estimates contain errors. We employ SVMs for classification, aiming to generalize well across different speakers. Results show that the proposed method improves the separation performance considerably compared to a previous CASA-based algorithm.

The paper is organized as follows. The proposed method is described in the following section. Evaluation and comparisons are given in Section 3, and we conclude the paper in Section 4.

## 2. SEPARATION OF UNVOICED SPEECH FROM VOICED SPEECH

### 2.1. Peripheral Processing

Cochannel speech is first decomposed by a 128-channel gammatone filterbank with center frequencies spread uniformly in the ERB (equivalent rectangular bandwidth) scale from 50 Hz to 8000 Hz [12]. The output of each filter is then divided into 20-ms time frames

with 10 ms overlap between neighboring frames. The resultant T-F representation is called a cochleagram [12]. In this representation, each T-F unit contains a filtered signal within a specific time frame and frequency band. To determine the UV intervals, we use an HMM-based pitch tracker [13] to estimate pitch. The UV intervals correspond to the frames with only one pitch estimate.

### 2.2. Feature Extraction

Since voiced speech is periodic (or quasi-periodic) and unvoiced speech is aperiodic, an important cue to differentiate voiced and unvoiced speech is pitch. Denoting a T-F unit at channel $c$ and frame $m$ as $u_{c,m}$, we extract the pitch-based 6F features following [6]

$$\mathbf{x}_{c,m} = \begin{pmatrix} A(c,m,\tau_m) \\ \mathrm{int}(\overline{f}(c,m)\cdot\tau_m) \\ |\overline{f}(c,m)\cdot\tau_m - \mathrm{int}(\overline{f}(c,m)\cdot\tau_m)| \\ A_E(c,m,\tau_m) \\ \mathrm{int}(\overline{f}_E(c,m)\cdot\tau_m) \\ |\overline{f}_E(c,m)\cdot\tau_m - \mathrm{int}(\overline{f}_E(c,m)\cdot\tau_m)| \end{pmatrix} \quad (1)$$

where $A(c,m,\tau_m)$ is an autocorrelation function [12] for $u_{c,m}$ at the time lag of $\tau_m$, which is the estimated pitch period at frame $m$. $A(c,m,\tau_m)$ measures the similarity between the detected period of the filtered signal in $u_{c,m}$ and $\tau_m$. $\overline{f}(c,m)$ is the estimated average instantaneous frequency of the filtered signal within $u_{c,m}$ [6]. The function $\mathrm{int}(x)$ rounds $x$ to the nearest integer. The product $\overline{f}(c,m)\cdot\tau_m$ thus provides another feature to measure the periodicity of $u_{c,m}$. It will be close to an integer greater or equal to 1 if the response in $u_{c,m}$ has a period of $\tau_m$. The third feature measures the deviation of $u_{c,m}$ from its nearest harmonic. The last three features (indicated by the subscript $E$) are extracted similar to the first three but from response envelopes [6].

$\mathbf{x}_{c,m}$ is extracted based on a single T-F unit. To capture temporal and spectral dynamics, we also calculate its delta features. For temporal dynamics, we calculate $\Delta\mathbf{x}^T_{c,m}$ by taking the element-wise difference between $\mathbf{x}_{c,m}$ and $\mathbf{x}_{c,m-1}$ when $m \geq 2$, and set $\Delta\mathbf{x}^T_{c,1}$ to be $\Delta\mathbf{x}^T_{c,2}$. The spectral deltas $\Delta\mathbf{x}^S_{c,m}$ are calculated in a similar way across frequency channels. The complete set of pitch-based features are thus

$$\mathbf{x}'_{c,m} = \begin{pmatrix} \mathbf{x}_{c,m} \\ \mathbf{x}^T_{c,m} \\ \mathbf{x}^S_{c,m} \end{pmatrix}. \quad (2)$$

The dimensionality of $\mathbf{x}'_{c,m}$ is $6 \times 3 = 18$. Feature extraction in (1) requires pitch. For training, we use Praat [14] to detect pitch from premixed clean utterances. In testing, pitch is estimated by an HMM-based pitch tracker [13] from the cochannel speech.

To further capture speech characteristics, we extract GFCC features for $u_{c,m}$. Note that the GFCCs here are extracted for each T-F unit instead of each frame as done in [11]. Specifically, for each channel, the filtered signal is filtered again by a 64-channel gammatone filterbank constructed as described in Section 2.1 for the 128-channel filterbank. The 64-channel outputs are then full-wave rectified, downsampled to 100 Hz along the time dimension, and compressed by a cubic root operation. Thus, for each T-F unit, we have a 64-dimensional spectral vector called a gammatone feature (GF) [11]. The GF is then converted to a GFCC by a discrete cosine transform [11]. As in [11], we take the first 31 cepstral coefficients and denote the GFCC for $u_{c,m}$ as $\mathbf{y}_{c,m}$.

To capture the temporal dynamics of $\mathbf{y}_{c,m}$, we use a delta-filter as constructed by [15]. The filter centers on the current T-F unit and spans across 9 frames. The output temporal delta feature is denoted as $\mathbf{y}^T_{c,m}$. Due to the high dimensionality of the GFCC features, we choose not to use the spectral delta features. Thus, the complete feature set of GFCC features are

$$\mathbf{y}'_{c,m} = \begin{pmatrix} \mathbf{y}_{c,m} \\ \mathbf{y}^T_{c,m} \end{pmatrix} \quad (3)$$

The dimension of the GFCC with temporal delta features (GFCC_D) is thus $31 \times 2 = 62$.

Combined with the pitch-based features, we thus have an 80-dimensional feature vector

$$\mathbf{z}_{c,m} = \begin{pmatrix} \mathbf{x}'_{c,m} \\ \mathbf{y}'_{c,m} \end{pmatrix} \quad (4)$$

for each T-F unit. Besides the pitch-based features and GFCC features, we have also considered the AMS features and their temporal and spectral deltas (AMS_DD) used in [10]. As shown in Section 3, the AMS_DD does not perform as well as the GFCC_D as a single feature type in classification. Further, we have tested the performance of adding the AMS_DD features to the current feature set but did not significantly improve performance. Considering the already high dimensionality of $\mathbf{z}_{c,m}$, we do not use the AMS_DD features for classification.

### 2.3. SVM-based Classification

We choose SVMs to train a speaker-independent classifier for separation. The SVMs maximize the distance between the samples of two classes near the separating hyperplane and are expected to generalize well [16].

We use the SVM for a linearly nonseparable scenario to perform separation [16]. In this case, the SVM is trained by minimizing the following cost function

$$f(\mathbf{w},\xi) = ||\mathbf{w}||^2/2 + C\sum_i \xi_i \quad (5)$$

with the constraints

$$y_i(\mathbf{w}'\mathbf{\Phi}(\mathbf{z}_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0. \quad (6)$$

In (5), $\mathbf{w}$ represents the weight vector of the separating hyperplane and $\xi_i$ is a nonnegative slack variable corresponding to the classification error. $C$ controls the tradeoff between the margin of two classes and the separation errors. In (6), $\mathbf{\Phi}$ is a mapping function projecting the training features to a higher-dimensional space, $y_i$ is the label for $\mathbf{z}_i$, and $b$ is the bias. We use $'$ to denote transpose. We use the LIBSVM package [17] for SVM training and testing.

We train an SVM for each channel. Since our task is to separate unvoiced speech from voiced speech, T-F units in the UV intervals that are dominated by the unvoiced speaker (no pitch) have the desired labels of 1 and those dominated by the other speaker (pitched) have the desired labels of 0. In feature mapping, we choose the radial basis function $K(\mathbf{z}_i,\mathbf{z}_j) = \exp(-\gamma||\mathbf{z}_i - \mathbf{z}_j||^2)$ as the kernel. Training parameters, $C$ and $\gamma$, are determined by cross-validation for each channel. A 5-fold cross-validation is chosen to maintain a balance between performance and computational complexity.

In classification, the outputs of an SVM are typically binarized by taking the signs. However, in our case, the two classes of training samples are unbalanced. For example, voiced speech is concen-

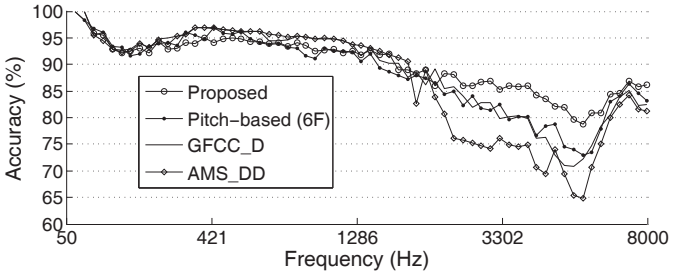


**Fig. 1**. Classification accuracies as a function of channel center frequency using different types of features. For each cochannel speech signal, we calculate the percent of correctly labeled T-F units for each channel. The average accuracy over all testing signals is shown above.

trated in low frequencies and thus there are more voiced-dominant T-F units. Similarly, more unvoiced-dominant T-F units can be found at high frequencies. As indicated in [10], this may cause the SVM hyperplane to skew to the minority class. To compensate for this imbalance, we use a development set to search for a threshold to binarize SVM outputs to maximize the classification accuracy for each channel. The new thresholds are used for binarization of SVM outputs in testing.

By combining all unvoiced-dominant T-F units in UV intervals we obtain a mask corresponding to the segregated unvoiced speech.

## 3. EVALUATION AND COMPARISON

To cover different speaker characteristics, we randomly choose 50 target speakers and another 50 interfering speakers from the TIMIT corpus [18] for training. To create cochannel speech signals, the 50 target speakers are randomly matched to the 50 interferer speakers to create 50 speaker pairs. For each pair, 10 utterances of the target are randomly matched to the 10 utterances of the interferer to create 10 0-dB cochannel speech signals. In mixing, the interfering signal is either extended or truncated to match the length of the target. In total, we have 500 mixtures for training. For this set, 25 mixtures are used for development and the others for SVM training. For testing, we randomly choose another 12 speakers and generate 60 0-dB cochannel speech mixtures similarly. Note that speakers used in testing are different from those in training.

We evaluate the segregation performance of our system based on the SNR gain of the segregated unvoiced speech. The SNR gain is calculated as the output SNR subtracted by the input SNR in the UV intervals. In computing the output SNR, we take the resynthesized speech from the UV portions of the IBM as the ground truth and measure the output SNR as

$$\mathrm{SNR} = 10\log_{10}(\sum_n S_I^2[n]/\sum_n (S_I[n]-S_E[n])^2), \qquad (7)$$

where $S_I[n]$ and $S_E[n]$ are the unvoiced speech signals resynthesized from the ideal and an estimated unvoiced IBM, respectively. In estimation, the UV intervals are determined by estimated pitch. For the IBM, ideal pitch contours extracted from premixed utterances by Praat [14] are used. Here, the IBMs are created using an LC (local SNR criterion [12]) of 0 dB. In addition to the SNR measure, we also evaluate using the classification accuracy, which is computed as the percent of T-F units correctly labeled in UV intervals.

We first compare the T-F unit classification accuracies using different features as a function of frequency. As shown in Fig. 1, each curve represents the classification performance based on one type of feature. Comparing all single feature types, we observe that the 6F pitch-based features perform comparably with GFCC_D features. The accuracies obtained by either feature are above 90% in low frequencies and about 80% on average in high frequencies. Across all frequency channels, the accuracy is about 88.5% on average. The AMS_DD feature performs a little worse. In low frequencies, it achieves a comparable performance to the 6F and GFCC_D features but performs significantly worse in high frequencies. The average accuracy is about 87.2%, being 1.3% lower than the other single feature types. On the other hand, the proposed feature set performs significantly better than any single feature type, with an average accuracy of 90.1%. The improvement is about 1.6% over the best single feature type, and the gap is as large as 5% in high frequencies where unvoiced speech is concentrated. The better performance of the proposed feature may imply that the 6F and GFCC_D features are complementary. We have also tried to combine the 6F and AMS_DD features but the performance is worse than the proposed feature set. Note that here pitches are detected by the HMM-based algorithm [13]. We have also evaluated the results using ideal pitch contours detected from clean source utterances by Praat. In the ideal case, the 6F features perform as the best single feature type with an accuracy of 89.8%, while the accuracy of the proposed feature also increases to 92.5%.

We compare the unvoiced speech separation of our system to the tandem algorithm [6], which relies on the 6F pitch-based features for separation. For the tandem algorithm, it also includes an iterative stage to further improve segregation using neighborhood T-F information and pitch continuity. We first compare the two algorithms based on the pitch contours detected by the tandem algorithm. As shown in the Tandem row in Table 1, the proposed method outperforms the tandem algorithm by 2.4 dB in terms of SNR gain and 3.9% in classification accuracy. To improve the tandem algorithm, we use the HMM-based pitch tracker [13] for pitch detection and apply the detected pitch to initialize the tandem algorithm. By initialization we mean the initial pitch estimates in the tandem algorithm are replaced by the outputs of the HMM-based pitch tracker. As shown in the HMM+Tandem row in Table 1, the performances of both algorithms improve. The proposed algorithm still performs better by 2.8 dB in SNR gain and and 3.9% in accuracy. It is worth mentioning that the tandem algorithm we compare to here is the version with improved pitch estimates. The best results for the proposed algorithm are obtained by directly using the pitch estimates from the HMM-based pitch tracker. The results are shown in the HMM Pitch row in Table 1. In this case, we achieve an SNR gain of 16.9 dB in UV intervals and the classification accuracy is 86.5%. Compared to the original tandem algorithm, the improvement is 4.1 dB in SNR gain and 5.6% in accuracy.

We further evaluate the system using a hit minus false alarm (HIT-FA) rate since it is shown that the HIT-FA rate is a good indicator of human speech intelligibility [9]. As in [9], we calculate the hit (HIT) rate as the percent of correctly labeled unvoiced-dominant T-F units and the false alarm (FA) rate as the percent of incorrectly labeled ones. The comparison with the tandem algorithm is presented in Table 2. In all pitch types, we observe that the proposed algorithm performs uniformly better than the tandem algorithm in terms of HIT-FA rates. Specifically, the proposed algorithm obtains HIT rates comparable to those of the tandem algorithm but achieves significantly lower FA rates. This probably indicates that the use of GFCC_D features complements the pitch-based features when there

**Table 2**. Comparisons in terms of HIT, FA and HIT-FA rates between the proposed method and the tandem algorithm

| Pitch type | HIT (%) | | FA (%) | | HIT-FA (%) | |
|---|---|---|---|---|---|---|
| | Proposed | Tandem | Proposed | Tandem | Proposed | Tandem |
| HMM Pitch | 60.0 | - | 7.3 | - | 52.7 | - |
| HMM + Tandem | 62.0 | 63.2 | 8.0 | 12.8 | 54.0 | 50.4 |
| Tandem | 60.1 | 60.7 | 9.3 | 14.1 | 50.8 | 46.6 |

**Table 1**. Comparisons in terms of average SNR gains and accuracies between the proposed method and the tandem algorithm

| Pitch type | SNR gain (dB) | | Accuracy (%) | |
|---|---|---|---|---|
| | Proposed | Tandem | Proposed | Tandem |
| HMM Pitch | 16.9 | - | 86.5 | - |
| HMM + Tandem | 16.1 | 13.3 | 86.6 | 82.7 |
| Tandem | 15.2 | 12.8 | 84.8 | 80.9 |

are miss errors in two-pitch frames. For our algorithm, the best HIT-FA rate is 54% when using the pitch from the HMM+Tandem case. It is 7.4% better than that of the original tandem algorithm. By using the HMM-based pitch alone, the HIT-FA rate is 52.7% and is 6.1% better than the original tandem algorithm. We note that these rates are generally lower than those in voiced portions since there are less speech-dominant T-F units in the UV portions. Further improvements can be obtained by rethresholding on the HIT-FA rate.

## 4. CONCLUSION

Inspired by classification-based methods for speech/nonspeech separation, we proposed an SVM-based classifier to separate unvoiced-voiced portions of cochannel speech. Different features are investigated for this task. We propose a new feature set combining pitch-based features and GFCC-based features and use channel-wise SVMs for classification. The proposed method is speaker independent, and results based on the TIMIT corpus show that it improves unvoiced speech segregation of a previous CASA-based algorithm.

## 5. ACKNOWLEDGMENT

This research was supported by an AFOSR grant (FA9550-08-1-0155).

## 6. REFERENCES

[1] G. Hu and D. L. Wang, "Segregation of unvoiced speech from nonspeech interference," *J. Acoust. Soc. Am.*, vol. 124, pp. 1306–1319, 2008.

[2] J. R. Hershey, S. J. Rennie, P. A. Olsen, and T. T. Kristjansson, "Super-human multi-talker speech recognition: a graphical model approach," *Comput. Speech Lang.*, vol. 24, pp. 45–66, 2010.

[3] A. Reddy and B. Raj, "Soft mask methods for single-channel speaker separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 6, pp. 1766–1776, 2007.

[4] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 1, pp. 1–12, 2007.

[5] K. Hu and D. L. Wang, "Unvoiced speech segregation from nonspeech interference via CASA and spectral subtraction," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, pp. 1600 – 1609, 2011.

[6] G. Hu and D. L. Wang, "A tandem algorithm for pitch estimation and voiced speech segregation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, pp. 2067–2079, 2010.

[7] D. L. Wang, "On ideal binary mask as the computational goal of auditory scene analysis," in *Speech Separation by Humans and Machines*, P. Divenyi, Ed., pp. 181–197. Norwell, MA: Kluwer Academic Press, 2005.

[8] Z. Jin and D. L. Wang, "A supervised learning approach to monaural segregation of reverberant speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, pp. 625–638, 2009.

[9] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou, "An algorithm that improves speech intelligibility in noise for normal-hearing listeners," *J. Acoust. Soc. Am.*, vol. 126, no. 3, pp. 1486–1494, 2009.

[10] K. Han and D. L. Wang, "An SVM based classification approach to speech separation," in *ICASSP-11*, 2011, pp. 4632–4635.

[11] Y. Shao and D. L. Wang, "Sequential organization of speech in computational auditory scene analysis," *Speech Comm.*, vol. 51, pp. 657–667, 2009.

[12] D. L. Wang and G. J. Brown, Eds., *Computational Auditory Scene Analysis: Principles, Algorithms and Applications*, Hoboken, NJ: Wiley-IEEE Press, 2006.

[13] Z. Jin and D. L. Wang, "HMM-based multipitch tracking for noisy and reverberant speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, pp. 1091–1102, 2011.

[14] P. Boersma and D. Weenink, "Praat: doing phonetics by computer (version 5.0.02)," Online: http://www.fon.hum.uva.nl/praat, 2007.

[15] D. P. W. Ellis, "PLP and RASTA (and MFCC, and inversion) in Matlab," Online: http://www.ee.columbia.edu/ dpwe/resources/matlab/rastamat/, 2005.

[16] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Min. Knowl. Discovery*, vol. 2, pp. 121–167, 1998.

[17] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," 2001.

[18] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT acoustic phonetic continuous speech corpus," 1993.