Recognition of Blowing Sound Types for Real-time Implementation in Mobile Devices

Marc-André Carbonneau,
Ghyslain Gagnon and Robert Sabourin
École de technologie supérieure
Department of Electrical Engineering
1100 Notre-Dame Ouest, Montréal, Canada

Jean Dubois
Université du Québec à Montréal
École des arts visuels et médiatiques
405 Ste-Catherine Est, Montréal, Canada

***Abstract*— This paper presents a system to recognize and classify sounds produced by human subjects blowing air by the mouth. The objective is to implement the system for fast recognition using low-complexity algorithms in a low-budget processor. Recognition is achieved using tailored band energy ratios, modified frequency centroid and a periodicity test based on spectrum autocorrelation. These lightweight feature extraction techniques are adapted to the particular task of recognition of blowing sound types. The classification is achieved by a naive Bayes classifier. The algorithm can be implemented in real-time (latency $\leq$ 100 ms) and experimental test results show average recognition rates over 94%.**

I. Introduction

Accurately identifying and classifying the different types of sounds that can be produced by a human blowing air by the mouth - for example, the sounds emitted when someone tries to cool-off or warm-up something - could be used in medical applications (e.g. control interface for the disabled), in video games and in children educational toys. In this case, it will be used in an interactive art piece to affect the multimedia content on an electronic display. For example, a user could blow towards a display and generate mist on a virtual window. Or, by blowing, a user could generate wind in a virtual environment. In this context, the system must continually recognize blowing sounds on small time windows (slices), in real-time. As long as the recognition rate is sufficiently high, erroneous decisions on single slices will have negligible effect on the rendered animation. On the other hand, the systems's latency is critical for an enjoyable experience for the user. Preliminary tests show that a latency lower than 100 ms is required with a recognition rate over 90%. Also, CPU usage must be minimized for integration in mobile systems and can thus be regarded as a problem of classification on a budget [1], [2].

While there are many effective sound and speech recognition techniques [3], [4], [5], [6], [7], very few information was found on their application to blowing sounds. Many of these techniques could succeed at this task, but they tend to rely on statistical methods that are CPU intensive or require buffering which precludes real-time classification.

This paper proposes lightweight feature extraction techniques which allow successful recognition of blowing sounds. These features are then classified by a standard naive Bayes classifier. The paper is organized as follows. In Section II, the characteristics of a blowing sound are described. In Section III, the feature extraction techniques are detailed. Section IV describes the classification. Section V describes the test methodology and the results are presented in Section VI, leading to a conclusion in Section VII.

II. Blowing Sounds Characteristics

In the study of voiced speech, the vocal system has been approximated efficiently by the independent source-filter model [8]. The vocal chords are modeled as a periodic sound source and the position of the parts of the mouth (e.g. lips, tongue, cheeks) acts as a variable filter, as shown in Fig. 1.

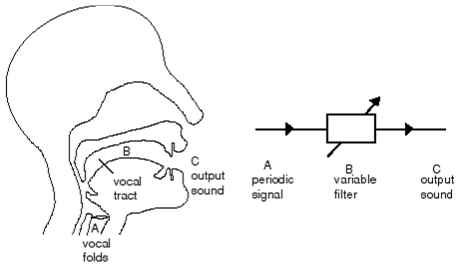

Fig. 1. Source-Filter Model. Reprinted with permission from [9].

When a person blows, the vocal chords are inactive. Only the position of the mouth affects the nature of the emitted sound. The emission of a blowing sound can thus be approximated by a white noise source passing through a bandpass filter. In this paper, two different blowing sound types are considered, labeled as hot and cold blowing sounds. A hot blowing sound is the type of sound made by someone trying to create mist on a window. A cold blowing sound is, for example, the sound of someone cooling-off a bowl of soup. Fig. 2 shows the power spectrogram of those two different blowing sound types, as well as speech, for comparison.

For both blowing sound types, the spectrum is not periodic and their energy is clearly located in different parts of the spectrum. A speech signal contains periodic components with fundamental frequencies comprised between 85 and 255 Hz [10].

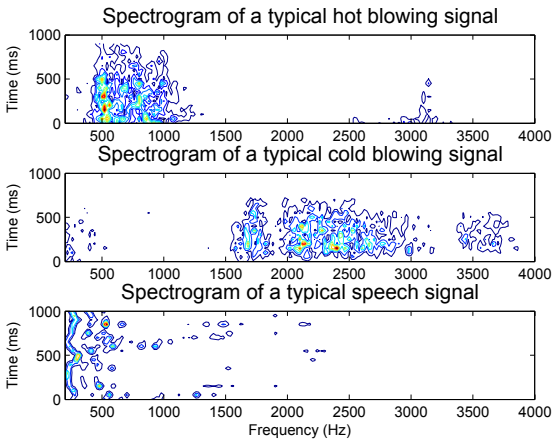

Fig. 2. Spectrograms for the three types of sound treated in this paper.

III. Features Extraction

In the proposed system, the incoming audio stream is segmented into uniform slices of length T_s with 50% interleaving. The Discrete Fourier Transform (DFT) of each slice is computed which is then used to obtain a number of signal features. These features are analyzed by the Bayes classifier to determine the sound type of each slice. Several signal features commonly used in sound or speech recognition, such as MFCC LPC or MP [11], [3], [4], [12], are too complex (i.e. computationnally expensive) to be implemented in a mobile device system for real-time implementation. The following feature extraction techniques have been selected for their efficiency and their low computation needs.

Band energy ratios (BER) (f_1-f_6) are used in the field of voice activity detection and speech recognition [7]. However, the technique has been adapted for the particular problem of blowing sound recognition. The spectrum of the signal is separated into four bands: [200 - 500 Hz], [500 - 1000 Hz], [1000 - 2300 Hz] and [2300 - 5000 Hz]. Each of the four bands contains frequencies specific to a recognized class of sound. The lowest band contains energy from speech. The second and third bands carry energy of a hot and cold blowing sound respectively. If a user is too close to the microphone, the signal is clipped and the fourth band carries energy from the third harmonic of the hot blowing sound. The energy in band x is obtained by integrating the DFT magnitude over its boundary frequencies $f_{min,x}$ and $f_{max,x}$:

$$\overline{A}_{sb_x} = \sum_{f=f_{min,x}}^{f_{max,x}} |X(f)| \tag{1}$$

The magnitude is not squared to reduce computation complexity.

A vector of $\binom{4}{2} = 6$ amplitude ratios r_{xy} is obtained from these 4 bands:

$$r_{xy} = \log \frac{\overline{A}_{sb_x}}{\overline{A}_{sb_y}} \tag{2}$$

The logarithm of these ratios is taken because it increases the recognition performances due to the large dynamic range of sound signals. These ratios provide more sensitive information than normalized power in each band because it emphasizes band power relation even if one band contains most of the spectrum power.

The spectral centroid (f_7) can be defined as the center of mass of the spectrum [13]. It is a simple yet powerful signal descriptor for this application. It was found that using only the frequencies for which the magnitude value is greater than 90% of the maximum magnitude significantly improves the reliability of the descriptor. The spectral centroid, f_c, is computed as follows:

$$f_c = \frac{\sum_{f=1}^{N} f|X'(f)|}{\sum_{f=1}^{N} |X'(f)|} \tag{3}$$

with

$$|X'(f)| = \begin{cases} 0 & \text{if } |X(f)| < 0.9X_{max} \\ |X(f)| & \text{if } |X(f)| \geq 0.9X_{max} \end{cases} \tag{4}$$

where X_{max} is the maximum observed magnitude.

The peak amplitude frequency (f_8) is simply the frequency at which the maximum magnitude X_{max} was observed. In most cases, this descriptor gives very similar results to the spectral centroid. However, it is simple to calculate and was shown to improve robustness when combined to the frequency centroid in noisy environments.

The last feature is a *periodicity evaluator* (f_9), which helps discarding speech signals because these signals contain significant energy in the frequency bands associated to blowing sounds. However, as stated in Section II, because the vocal cords are inactive when blowing, blowing sounds do not exhibit short-term periodicity. A periodicity test is therefore efficient at improving the recognition performance when speech sounds are injected in the system, which is to be expected in this application.

A low computation complexity periodicity test based on the spectrum autocorrelation was designed. When computing the autocorrelation of the DFT, a periodic signal will exhibit strong peaks representing the fundamental frequency sliding over its harmonics, as shown in Fig. 3 [14]. An algorithm was developed to search for peaks and then cumulates a score at each peak based on its amplitude and shape. This score is the value for feature f_9. Due to the length constraint of this paper, the algorithm cannot be further detailed here.

IV. Classification

There are three classes in this problem: cold blowing sounds, hot blowing sounds, and non-blowing sounds. In most cases non-blowing sound will be ambient noise or speech. Because of the large variability of non-blowing sounds, a class-modular approach [15] was adopted. Two naive Bayes classifiers [16] are implemented, one to discriminate hot blowing sounds from all other sounds, and another classifier to discriminate cold blowing sounds from all other sounds.

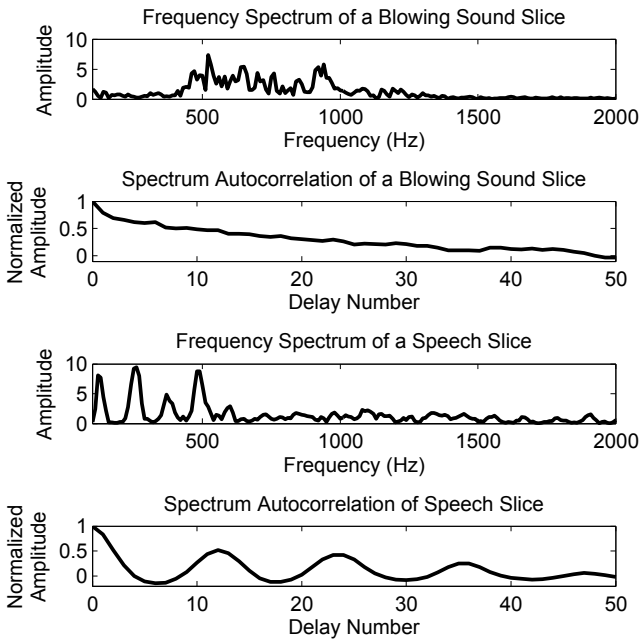

Fig. 3. Spectrum and spectrum autocorrelation

There are nine features in this problem which are described in Section III: the six band energy ratios, the spectral centroid value, the peak amplitude frequency and the periodicity score. As an approximation, these features are considered to be independent. This approximation simplifies the computation while providing acceptable results [16], [17].

A reference feature vector (T) is constructed for each class. Each vector component (t_i) contains the mean (μ_{t_i}) and variance $(\sigma^2_{t_i})$ computed from the training samples for the nine features. The feature vector (F) of the sample under test is compared to the reference vector (T). The features probability density functions (PDF) are evaluated assuming a Gaussian distribution:

$$p\left(f_i|t_i\right) = \frac{1}{\sqrt{2\pi\sigma^2_{t_i}}} e^{\frac{-\left(f_i-\mu_{t_i}\right)^2}{2\sigma^2_{t_i}}} \tag{5}$$

The Kolmogorov-Smirnov test showed that the features PDF is not strictly Gaussian, meaning that the system is not optimal. However, the cumulative distribution function (CDF) of each feature, shown in (Fig. 4), gives insight as to why the system provides good performances under the assumption of a Gaussian distribution.

Assuming all features are independent, the probability of the slice being part of a class (C) is [16]:

$$p(C|f_1,...,f_9) = Kp(C)\prod_{i=1}^{9} p\left(f_i|t_i\right) \tag{6}$$

where K represents the evidence on Bayes' Theorem. It is constant for each class and thus can be ignored in the implementation to save computation. Moreover, the prior

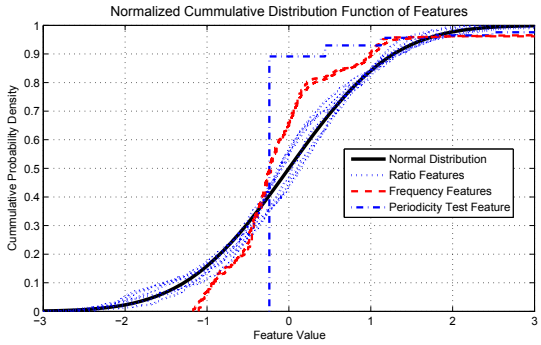

Fig. 4. CDF of all features, as well as normal CDF for comparison.

probabilities $p(C)$ are unknown for this application; each class is thus considered equiprobable and the scaling factor $p(C)$ is not computed to save a multiplication in the hardware implementation.

Each sound slice is classified as either a hot or cold blowing sound based on the highest probability as computed in (6) for each class. However, if this probability is below a certain threshold in both classes, the sound slice is classified as a 'non-blow' signal.

V. Test Methodology

To create a sample database, six different subjects - three women and three men - were recorded while emitting blowing sounds. Proper care was taken to avoid saturation of the microphone but no precautions were taken to minimize ambient noise. To emit hot blowing sounds, the test subjects were asked to blow as if they were trying to create mist on a window. They were instructed to keep their mouth wide open while blowing. To emit cold blowing sounds, the subjects were asked to blow as if they were trying to cool-off a bowl of soup. Finally, they were asked to speak normally in the microphone to create speech samples.

This session resulted in 99 different recorded samples of hot sounds, 75 samples of cold sounds and 48 samples of speech sounds. The attack and release parts of the sounds were removed from the samples. Each sample was divided into 50% interleaving slices of duration T_s. Generally, the recognition rate increases with T_s, at the price of a higher latency. Hence, the slices length was fixed at $T_s = 200$ ms due to the tolerable latency of 100 ms. This created 1000 slices of hot blowing sounds, 1300 slices of cold blowing sounds and 389 slices of speech signals. The proposed algorithm was implemented in MATLAB with the the recorded .wav files being treated as an incoming audio stream (no look ahead).

To test the system, 10 samples of hot sound and 10 samples of cold samples were randomly selected. All remaining samples were used to train the classifier by obtaining the mean and variance of each feature for both classes. The 20 selected blowing sound samples and every speech samples were then classified. This test was repeated 100 times. This is called the "jackknife" testing method and has been used in previous sound recognition works [4], [6].

VI. Results

Table I shows the classification performances when using slices of duration T_s = 200 ms, as explained in Section V. The average recognition performance is 93% or better for each sound type. It must be emphasized that these results represent the recognition rate of every 200 ms sound slice. In other words, for the duration of a full hot blowing sound, the classifier would be at the correct value for 93% of the time.

TABLE I
Classifier results for 200 ms slices

	Classified as:		
Slice type	Hot	Cold	Non-Blow
Hot	93.0 ± 6.0 %	2.3 ± 2.5 %	4.7 ± 5.4 %
Cold	3.9 ± 4.5 %	95.1 ± 4.4 %	1.0 ± 1.2 %
Speech	3.7 ± 0.4 %	0.5 ± 0.0 %	95.8 ± 0.4 %

TABLE II
Classifier results for 100 ms slices

	Classified as:		
Slice type	Hot	Cold	Non-Blow
Hot	93.5 ± 3.5 %	2.2 ± 1.7 %	4.3 ± 3.2 %
Cold	5.0 ± 3.1 %	94.0 ± 3.3 %	1.0 ± 1.1 %
Speech	5.0 ± 0.2 %	0.5 ± 0.0 %	94.5 ± 0.3 %

Tests using a smaller slice length (100 ms) result in slight degradation of classification performances (see Table II). However, the response time of the system is halved. For the same full length sample, there will be twice as much slices analysed. The number of data (N) in each of these slice will be scaled by a factor of two. Because FFT and autocorrelation algorithms complexity is $O(Nlog_2N)$, the number of operations needed to achieve classification is reduced. Also, less memory is used during the process.

Assuming each class is equiprobable in the application (the probability will be assessed in *in situ* experiments with the implemented system), the average performance for the recognition system is 94.6%, which is better than the 90% performance target.

VII. Conclusion

In this work, a system to recognize two different types of blowing sounds was presented. The system can be implemented on an mobile device platform for real-time operation ($\leq$100 ms latency) and the classification can be achieved by a classic naive Bayes classifier. We identified and optimized features that allow high recognition rates of blowing sound types with low computation complexity. These features are frequency band energy ratios, the spectral centroid, the peak amplitude frequency of the spectrum and a periodicity test score. The frequency centroid feature has been made more discriminative by introducing amplitude thresholding. Also a periodicity test based on the spectrum autocorrelation was proposed.

Future work includes implementation of the algorithm in an Android system and rigorous validation with a higher number samples from different test subjects. It also includes the design of a tracking algorithm which would take into account previous decisions of the classifier to improve the recognition performances. The performances could also be improved by training the system for a specific user at the beginning of a session. Finally, tests should be conducted to see if the classifier could benefit from the implementation of new classes, e.g. separate classes for male and female users.

References

[1] K. Singer, "Online classification on a budget," in *Advances in neural information processing systems 16: proceedings of the 2003 conference*, vol. 16. The MIT Press, 2004, p. 225.

[2] J. Weston, A. Bordes, and L. Bottou, "Online (and offline) on an even tighter budget," in *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics, January 2005, Barbados*, R. G. Cowell and Z. Ghahramani, Eds. Society for Artificial Intelligence and Statistics, 2005, pp. 413–420.

[3] M. Cowling and R. Sitte, "Recognition of environmental sounds using speech recognition techniques," *Advanced signal processing for communication systems*, pp. 31–46, 2002.

[4] ——, "Comparison of techniques for environmental sound recognition," *Pattern Recognition Letters*, vol. 24, no. 15, pp. 2895–2907, 2003.

[5] A. Eronen, V. Peltonen, J. Tuomi, A. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, "Audio-based context recognition," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 321–329, 2006.

[6] R. Goldhor, "Recognition of environmental sounds," in *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, vol. 1. IEEE, 1993, pp. 149–152.

[7] V. Peltonen, J. Tuomi, A. Klapuri, J. Huopaniemi, and T. Sorsa, "Computational auditory scene recognition," in *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE Intl. Conf. on*, vol. 2. IEEE, 1993, pp. II–II.

[8] G. Fant, *Acoustic theory of speech production*. Walter de Gruyter, 1970, no. 2.

[9] J. Wolfe, M. Garnier, and J. Smith, "Vocal tract resonances in speech, singing, and playing musical instruments," *HFSP journal*, vol. 3, no. 1, pp. 6–23, 2009.

[10] R. Baken and R. Orlikoff, *Clinical measurement of speech and voice*. Singular Pub Group, 2000.

[11] S. Chu, S. Narayanan, and C. Kuo, "Environmental sound recognition with time–frequency audio features," *Audio, Speech, and Language Processing, IEEE Trans. on*, vol. 17, no. 6, pp. 1142–1158, 2009.

[12] O. Uribe, H. Meana, and M. Miyatake, "Environmental sounds recognition system using the speech recognition system techniques," in *Electrical and Electronics Engineering, 2005 2nd International Conference on*. IEEE, 2005, pp. 13–16.

[13] B. Iser, W. Minker, and G. Schmidt, *Bandwidth extension of speech signals*. Springer Verlag, 2007, vol. 13.

[14] H. K. Kim and H. S. Lee, "Use of spectral autocorrelation in spectral envelope linear prediction for speech recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 7, no. 5, pp. 533–541, 1999.

[15] I.-S. Oh and C. Y. Suen, "A class-modular feedforward neural network for handwriting recognition," *Pattern Recognition*, vol. 35, no. 1, pp. 229–244, Jan. 2002.

[16] P. Domingos and M. Pazzani, "On the optimality of the simple bayesian classifier under zero-one loss," *Machine learning*, vol. 29, no. 2, pp. 103–130, 1997.

[17] D. Hand and K. Yu, "Idiot's bayes-not so stupid after all?" *International Statistical Review*, vol. 69, no. 3, pp. 385–398, 2001.