SOLAR: Sound Object Localization and Retrieval in Complex Audio Environments

Derek Hoiem[1], Yan Ke[1], Rahul Sukthankar[2,1]

[1] School of Computer Science, Carnegie Mellon University; [2] Intel Research Pittsburgh
{dhoiem, yke, rahuls}@cs.cmu.edu
http://www.cs.cmu.edu/~dhoiem/projects/solar

ABSTRACT

The ability to identify sounds in complex audio environments is highly useful for multimedia retrieval, security, and many mobile robotic applications, but very little work has been done in this area. We present the SOLAR system, a system capable of finding sound objects, such as dog barks or car horns, in complex audio data extracted from movies. SOLAR avoids the need for segmentation by scanning over the audio data in fixed increments and classifying each short audio window separately. SOLAR employs boosted decision tree classifiers to select suitable features for modeling each sound object and to discriminate between the object of interest and all other sounds. We demonstrate the effectiveness of our approach with experiments on thirteen sound object classes trained using only tens of positive examples and tested on hours of audio data extracted from popular movies.

1. INTRODUCTION

The ability to identify sound events allows humans to listen for cars while crossing the road, notice when the dog barks to come in, and infer from the metallic clang that Aragorn successfully deflects the Uruk-Hai's thrown dagger in the film *The Fellowship of the Ring*. Similarly, the development of sound identification systems is crucial to many applications, including multimedia retrieval, security, and mobile robotics. With the ability to automatically identify sounds, viewers could search for the action sequences in their DVD collections, security infrastructures could detect gunshots or cries for help, and mobile robots could better understand their environment.

Despite the breadth and importance of these applications, little research has been done to enable the identification of sounds in complex environments. Much research has been directed towards classifying a short sound clip into one of a pre-specified set of categories [1, 4, 6, 7, 11]. This work is useful for organizing databases of sound clips but cannot be directly applied to the task of sound detection and identification in complex audio environments. To detect a particular class of sound objects, such as dog barks, it is necessary to distinguish instances of that class from *all* other possible sounds. These categorization systems, however, can only distinguish among a highly finite set of sound categories. Additionally, in order to avoid hundreds of false positives per hour, the sound object detection system must achieve false positive rates (the percentage of incorrectly labeled non-object sounds) of well under 1%. Current sound categorization systems, however, typically achieve error rates of between 5% and 20%.

Few systems have been developed for the detection and identification of sound objects, and these generally assume unrealistic conditions. The system proposed by Dufaux et al. [3] achieves excellent results under a white noise background but fails under real-world conditions. Zhang and Kuo [12, 13] segment audio into voice, music, or environmental sound and categorize environmental sound segments, but their system has trouble segmenting audio in which sounds co-occur with music or voice and can only classify among a set of ten narrow categories.

2. SOUND DETECTION

Our goal is to localize and retrieve sound objects such as gunshots, dog barks, laughter, sword clashes, laser guns, and screams that correspond to particular action events. This task is difficult due to within-class variance, background noise, and the large number of sounds that could potentially be confused with the object of interest. For examples of in-class variance, contrast the laugh of Vincent Price with that of Pee-Wee Herman or the shrill yap of the Chihuahua with the deep bark of the German Shepherd. In complex audio environments, such as in the audio data from movies, background noise is continually present and can often drown out the sound object of interest [2]. Since the sound objects occupy a tiny portion of the overall audio data, the data must either be segmented along boundaries of sound events or evaluated in short windows at every possible time location within the data. In either case, the sound detection system must be able to discriminate between the sound object of interest and all

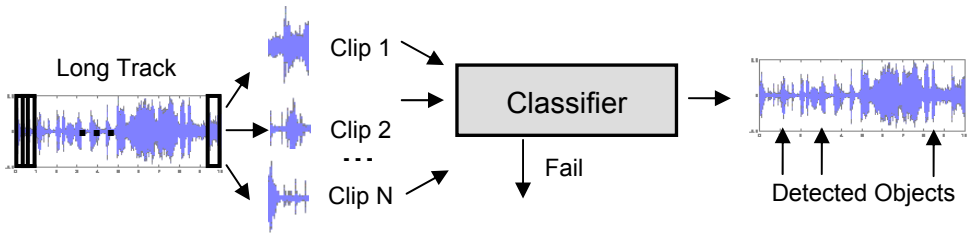

Figure 1. SOLAR avoids segmentation by decomposing audio tracks into short, equal-length, overlapping audio windows. Each window is evaluated by a boosted decision tree classifier. Finally, sound clips containing audio identified as belonging to the object class are returned to the user, sorted by classifier confidence.

Figure 2. Raw audio data (a) of a meow is converted into a rich feature representation in three steps: (b) Compute the short-time Fourier transform (STFT) from 200 Hz to 4 KHz and separate into 16 frequency channels, (c) Represent the STFT as the total power over time (bottom) and the percentage of power in each frequency channel over time (top), (d) Extract features representing pitch, loudness, and many other audio characteristics from representation in (c).

other sounds. Since the number of non-interesting sounds by far outnumbers the number of interesting sounds, a low false positive rate is critical.

We propose the SOLAR system as a solution to sound identification in complex audio environments. We avoid unreliable segmentation by performing a windowed scan over the audio data. The windowed scan involves sliding a window over the audio data in fixed increments and classifying the data contained in each window (see figure 1). For instance, if attempting to detect gunshots, the system would evaluate audio windows of one half-second duration in increments of one-sixteenth of a second, resulting in 57,600 windows per hour of audio data. A classifier with a detection rate of 80% and a false positive rate of 1% searching over an hour of audio data containing five gunshots would retrieve 580 audio clips, only four of which would contain the object of interest. Thus, highly selective classifiers achieving false positive rates of well under 1% are necessary for the system to be useful.

SOLAR achieves high detection rates and low false positive rates for many different sound objects by using a diverse feature set and boosted decision tree classifiers. The features, based on the short-time Fourier transform of the data, represent perceptual characteristics such as pitch and loudness and non-perceptual information such as the approximate band-width of the audio.

Each audio window is classified as being the object of interest if its confidence, as assigned by a learned classifier for that object, exceeds a given threshold. The classifier is composed of a series of boosted decision trees. Decision trees greedily select the best features for discrimination and make decisions based on those features. By assigning a confidence to each decision and using Adaboost [9] to re-weight training data after learning each decision tree, a weighted vote may be obtained from all of the decision trees that is more accurate than any single decision tree.

3. REPRESENTATION

The feature representation needs to be diverse and discriminative enough to allow the classifiers to distinguish between any of a large variety of sound classes and all other sounds. Many researchers in sound categorization have found perceptual features such as pitch and loudness to be useful [11], and others have shown that a mixture of perceptual and non-perceptual features may provide the best classification performance [4]. Based on the short-time Fourier transform (STFT), we provide our classifiers with a rich representation capable of modeling a variety of classes and useful for discrimination (see figure 2).

At the most basic level, we represent audio with the short-time Fourier transform using hamming windows of 128 ms spaced every 12.5 ms. We then divide the frequencies from 200 Hz to 4 KHz into 16 channels. We choose the ranges of the channels such that the average power for each channel is approximately equal for typical audio data from movies. These learned range boundaries are {200, 240, 290, 350, 410, 480, 580, 670, 780, 870, 1000, 1170, 1320, 1530, 1870, 2510, 4000} Hz. This channeling has higher resolution in the voice range of 400-1800 Hz and lower resolution in the high frequency range than the common logarithmically-spaced channeling. After computing the STFT, we divide each frequency-time coefficient by the total power at that time. Our representation then becomes a 16xT set of coefficients representing the percent of power in each frequency channel over time and a 1xT set of coefficients representing the total power over time, where T is the number of Fourier windows in

the audio data. All features used by the classifiers are computed from this 17xT representation.

We chose the candidate set of 138 features for their computational efficiency and for their ability to represent distinguishing characteristics of a broad range of sounds. Features encode the mean and standard deviation of the power percentile in each frequency channel and of the total power, bandwidth, the most powerful frequency channel, the number of peaks in power over time, the regularity of power peaks, the range of the total power over time, and time-localized frequency percentiles over various frequency ranges. Some of these features, such the mean of the power percentile of the frequency channels, capture basic audio characteristics that are useful to model for all sounds. Others, such as the range in total power over time, are more relevant to some sounds than others. Details of the feature representation are available at our web site.

4. CLASSIFIERS

SOLAR's classifiers are able to choose an appropriate feature representation from the set of 138 candidate features and to use those features to discriminate between the sound object of interest and all other sounds. SOLAR's high classification performance is achieved through the use of two classification techniques: the use of decision trees to select discriminating features and the use of Adaboost to improve classification with an ensemble of trees.

4.1. Decision Tree Classifiers

Decision tree classifiers are used to select features capable of discriminating between the object of interest and all other sounds [8]. The decision tree is formed by greedily choosing the most discriminative feature and making a decision based on that feature. The tree then continues to branch out, adding new features at the leaf nodes in order to greedily maximize the split between the object and non-object class until all training examples are correctly classified. Typically, the tree is then pruned to improve the generalization ability of the classifier. In forming the decision trees, we use the Gini diversity index for the splitting criterion and prune the trees to minimize the cross-validation error.

4.2. Boosting the Classifiers

A single decision tree may not be complex enough to model the differences between objects and non-objects without overfitting. We, therefore, use a standard machine learning technique called Adaboost [9] to improve classification accuracy. Using Adaboost, we iteratively learn a series of decision tree classifiers that focus on the mistakes of the previously learned classifiers. Confidences are assigned to the decisions made by each classifier according to the class-conditional log likelihood ratio, and the final confidence assigned to the label of an audio window is given by the sum of the individual classifier confidences. When evaluating new audio data, windows are sorted by confidence so that the user hears the sound clips most likely to contain the sound object interest before hearing less likely sound clips. A final classification can be assigned based on a confidence threshold. Using validation data, this threshold can be chosen to correspond to a particular false positive rate.

5. EXPERIMENTS

In our experiments, we attempted to learn models capable of identifying thirteen different sound object classes: car horns, doors closing, dog barks, door bells, explosions, gunshots, laser guns, light sabers, male laughs, meows, telephone rings, screams, and sword clashes. These sounds varied widely, from being impulsive (gunshots) to nearly monotone (meows) and from periodic (telephone rings) to non-periodic (doors closing) and from low frequency (explosions) to high frequency (sword clashes). We collected roughly 15-80 positive examples for each sound class using www.findsounds.com. About 1,000 negative training examples for each object were randomly drawn from audio data from movies that did not contain the sound of interest. All positive examples were mixed with background noise taken from the movie audio so that 5-25% of the amplitude was due to noise. Additionally, positive examples that were shorter in duration than the window size (fixed according to the median length of object training examples) were embedded into background noise sampled the movie audio. Half of the positive examples were used for training. The test data was composed of the remainder of the positive examples and about one hour of audio per object from movies not used for training. Detection rates were calculated based on the confidences assigned to the object examples, and false positive rates were calculated based on the number of detections in the non-object movie audio data.

5.1. Feature Usage

If a simple feature representation is sufficient to represent many different sound objects, we would expect the same features to be used consistently throughout the models for the different classes. If, however, having a rich set of diverse features is important, we would expect different sound objects to use different features. The mean number of features used for classification per sound object was 45.9 with a standard deviation of 14.8. Altogether, 133 out of the 138 features were used in at least one of the thirteen sound object classes. Thus, nearly all of the features were useful for at least one class, and different

	aver-age	car horn	close door	dog bark	door bell	explo-sion	gun-shot	laser gun	light saber	male laugh	meow	phone ring	scream	sword clash
10 FP/hr	37%	46%	0%	31%	81%	0%	0%	31%	52%	7%	86%	65%	38%	47%
50 FP/hr	60%	66%	22%	67%	90%	25%	0%	67%	81%	23%	92%	96%	83%	63%
100 FP/hr	72%	76%	32%	85%	99%	50%	21%	80%	83%	45%	97%	96%	89%	88%

Table 1. Sound object detection rates for varying numbers of false positives per hour of audio test data. For instance, a user willing to accept ten false positives per hour of audio could expect SOLAR to find five barks in an audio clip containing fifteen dog barks.

features were useful for different classes, indicating the importance of having a rich feature set.

5.2. Detection Time

SOLAR evaluates audio at a rate of roughly 24 times real-time (or about 2.5 minutes per hour of audio data) when running in MATLAB® on an Intel® Pentium® IV 3.2 GHz machine. This is fast enough for real-time security or robotic applications and with optimizations would be fast enough for useful multimedia retrieval. Additionally, 96% of the processing time is spent on computing features, so, if the features were pre-computed, SOLAR could run at 600 times real-time (5 seconds per hour of audio data). The system can also take advantage of database systems recently developed to allow fast searches over non-indexed data [5].

5.3. Detection Accuracy

Table 1 presents the chance of SOLAR identifying one particular instance of a sound object if 10, 50, or 100 false positives per hour of audio data are allowed. We use this form of displaying results because it reflects the user's experience. The reader should keep in mind that for each hour of audio data, 30,000-60,000 audio windows need to be evaluated. Thus, ten false positives per hour is equivalent to a false positive rate of only 0.0002 to 0.0003. For qualitative results of retrieved sounds for different sound objects from movie clip audio data, the reader should visit our web site.

6. CONCLUSIONS

We have presented SOLAR, the first system capable of finding sound objects in complex audio environments. SOLAR uses a windowed scan to eliminate the need for segmentation and employs boosted decision tree classifiers to achieve excellent performance on many sound objects. While SOLAR is able to achieve good results on many objects, it performs quite poorly on others such as door slams or male laughs (male laughs are often confused with male dialogue). Much of this error is due to lack of context, which could be acquired from other modalities. We hope to extend SOLAR to employ audio information in conjunction with visual scene information.

7. ACKNOWLEDGEMENTS

The authors would like to thank Roger Dannenberg for useful feedback and the anonymous reviewers for helpful comments and suggestions. Derek Hoiem and Yan Ke were supported by an Intel Research internship for part of this research.

8. REFERENCES

[1] M. Casey, "Reduced-Rank Spectra and Minimum-Entropy Priors as Consistent and Reliable Cues for Generalized Sound Recognition," *Workshop for Consistent & Reliable Acoustic Cues*, 2001.

[2] L. Chen, S. J. Rizvi, and M. T. Özsu, "Incorporating Audio Cues into Dialog and Action Scene Extraction", *Proc. of SPIE Storage and Retrieval for Media Databases*, 2003.

[3] A. Dufaux, L. Besacier, et al., "Automatic Sound Detection and Recognition for Noisy Environment," *Proc. of the X European Signal Processing Conference*, 2000.

[4] G. Guo and S. Li, "Content-Based Audio Classification and Retrieval by Support Vector Machines," *IEEE Trans. on Neural Networks*, 2003.

[5] L. Huston, R. Sukthankar, et al., "Diamond: A Storage Architecture for Early Discard in Interactive Search", *USENIX Conf. on File and Storage Technologies*, 2004.

[6] G. Li and A. Khokhar, "Content-based Indexing and Retrieval of Audio Data using Wavelets," *IEEE Int. Conf. on Multimedia and Expo*, 2000.

[7] S. Li, "Content-Based Classification and Retrieval of Audio Using the Nearest Feature Line Method," *IEEE Trans. on Speech and Audio Processing*, 2000.

[8] J.R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

[9] R. Schapire and Y. Singer, **"**Improved boosting algorithms using confidence-rated predictions," *Machine Learning*, 1999.

[10] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *IEEE Conf. on Computer Vision and Pattern Recognition*, 2001.

[11] E. Wold, T. Blum, et al., "Content-Based Classification, Search, and Retrieval of Audio," *IEEE Multimedia*, 1996.

[12] T. Zhang and C. Kuo, "Content-Based Classification and Retrieval of Audio", *SPIE Conf. on Adv. Signal Processing Algorithms, Architecture and Implementations VIII*, 1998.

[13] T. Zhang and C. Kuo, "Hierarchical System for Content-based Audio Classification and Retrieval," *Proc. Of Int. Conf. on Acoustics, Speech, and Signal Processing*, 1999.