

19

Analysis and Synthesis of Sound Textures

Nicolas Saint-Arnaud and Kris Popat
The MIT Media Laboratory

The sound of rain or of a large crowd are examples of sound textures. A restricted definition of *sound texture* is proposed for machine processing. Sound textures are treated as two-level phenomena: simple sound elements called atoms form the low level, and the distribution and arrangement of atoms form the high level. A cluster-based probability model is used to characterize the high level of sound textures. The model is then used to resynthesize textures that are perceptually similar to originals (training data). Finally, applications of the model for classification of sound textures are suggested.

19.1 DEFINITION OF A SOUND TEXTURE

In this chapter we present a method for resynthesis of sound textures, like the sound of rain, large crowds, fish tank bubbles, photocopiers and myriad others.

Defining *sound texture* is no easy task. Most people will agree that the noise of a fan is a likely “sound texture.” Some other people would say that a fan is too bland, that it is only a noise. The sound of rain, or of a crowd are perhaps better textures. But few will say that one voice makes a texture.¹

19.1.1 First Constraint in Time: Constant Long-Term Characteristics

A definition for a sound texture could be quite wide, but we chose to restrict our working definition for many perceptual and conceptual reasons. First of all, there is no consensus among people as to what a sound texture might be; more people will accept sounds that fit a more restrictive definition.

The first constraint we put on our definition of a sound texture is that it should exhibit similar characteristics over time, that is, a two-second snippet of a texture should not differ significantly from another two-second snippet. A

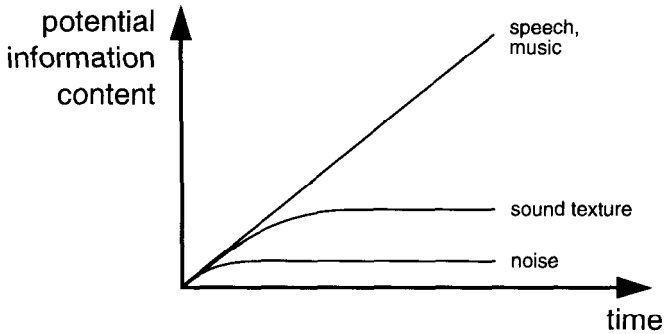


FIG. 19.1. Sound textures and noise show constant long-term characteristics.

sound texture is like wallpaper: it can have local structure and randomness, but the characteristics of the fine structure must remain constant on the large scale.

This means that the pitch should not change like that of a racing car, the rhythm should not increase or decrease, and so on. This constraint also means that sounds in which the attack plays a great part (like many timbres) cannot be sound textures. A sound texture is characterized by its sustain.

Fig. 19.1 shows an interesting way of segregating sound textures from other sounds, by showing how the “potential information content” increases with time. “Information” is taken here in the cognitive sense rather than the information theory sense. Speech or music can provide new information at any time, and their “potential information content” is shown here as a continuously increasing function of time. Textures, on the other hand, have constant long term characteristics, which translates into a flattening of the potential information increase. Noise (in the auditory cognitive sense) has somewhat less information than textures.

Sounds that carry a lot of meaning are usually perceived as a message. The semantics take the foremost position in the cognition, downplaying the characteristics of the sound proper. We choose to work with sounds which are not primarily perceived as a message, that is, nonsemantic sounds, but we understand that there is no clear line between semantic and non-semantic. Note that this first time constraint about the required uniformity of high level characteristics over long times precludes any lengthy message.

19.1.2 Two-Level Representation

Sounds can be broken down to many levels, from a very fine (local in time) to a broad view, passing through many groupings suggested by physical, physiological and semantic properties of sound. We choose, however, to work with only two levels: a low level of simple atomic elements distributed in time and a high level describing the distribution in time of the atomic elements.

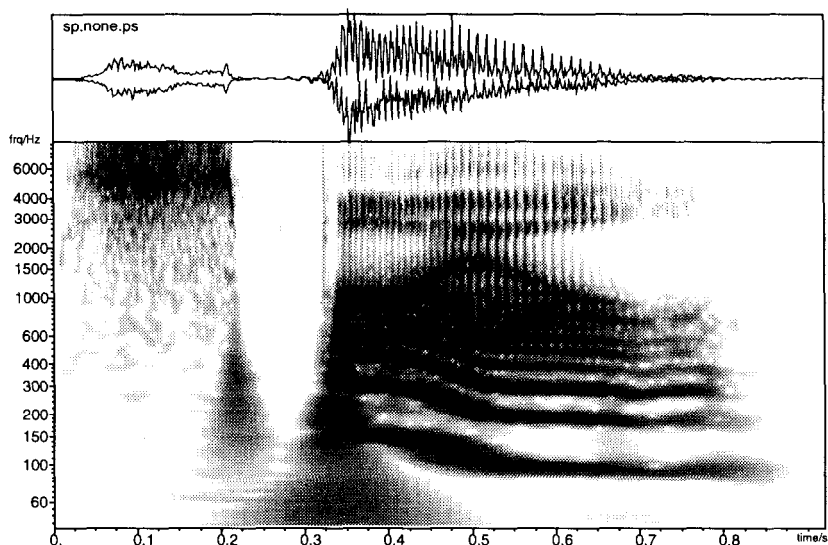


FIG. 19.2. Example of a time-frequency representation: a constant-Q transform of the word “spoil.” (Courtesy of Dan Ellis.)

For many sound textures - applause, rain, fish-tank bubbles - the sound atom concept has physical grounds. Many more textures can also be usefully modeled as being made up of atoms. Without assuming that all sounds are built from atoms, we use the two-level representation as a model for the class of sound textures that we work on.

The boundary between low and high level is not universal or fixed, and we will in fact move it, by sometimes using very primitive atomic elements, sometimes using more complex atoms. Note that using simpler atoms leaves the high level to deal with more information and more complexity. On the other hand, one should be careful not to make overly narrow assumptions — losing generality — when choosing more complex atomic elements.

Such a two-level representation has some physical grounding, as explored in “Auditory Perception of Breaking and Bouncing Events” (Warren and Verbrugge, 1988). In this paper, Warren and Verbrugge present a “structural” level characterized by the properties of the objects being hit, and a “transformational” level, characterized by the pattern of successive hits in breaking and bouncing events.

19.1.3 Low Level: Sound Atoms

The signal captured by a microphone is a time waveform, which can be digitally represented by Pulse Code Modulation (PCM) (Reddy, 1976). In the human ear, the cochlea performs a time-frequency transform. Fig. 19.2 shows the time waveform for an occurrence of the word “spoil”, and an example of time-

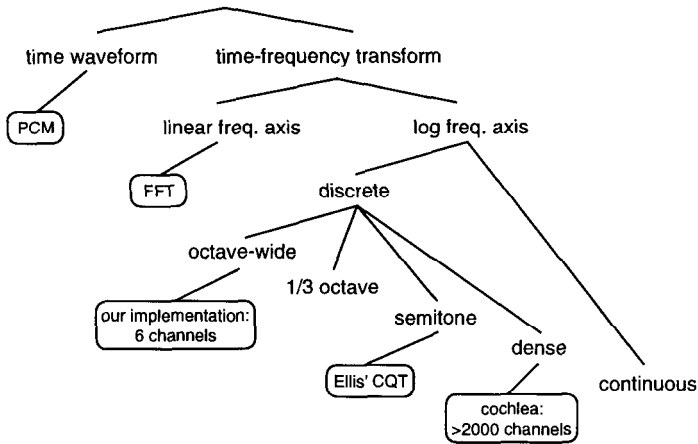


FIG. 19.3. A variety of sound representations.

frequency transform (a constant-Q transform) underneath. A time-frequency representation is often called spectrogram.

Fig. 19.3 shows a variety of sound representation domains, including the time waveform (PCM) and a few possible time-frequency diagrams.

Sound atoms usually form patterns in one of the representation domains. For example, the time-frequency representation of the phonemes of the word “spoil” on Fig. 19.2 could be used as atoms. The atom patterns can be complex, but we can also choose simpler atoms, such as groupings of energy continuous in time and frequency. Finally, the instantaneous energy in each frequency channel of a time-frequency transform form the simplest atomic features possible.

Instantaneous Energy. In our implementation for resynthesis, we choose to use the simplest atomic elements: instantaneous energy in one frequency channel. In our implementation, a straightforward filter bank splits the incoming sound in six octave-wide frequency bands. The energy levels in each band are our atomic “features.” The small number of frequency bands is convenient for computation, as we will see in the implementation.

In the limit, with one frequency band (no filtering), atoms can be chosen as the instantaneous PCM value in the broadband signal. This simplistic approach shifts the computational burden to the high level. The “two sines” resynthesis example in section 19.3.3 uses the PCM values as atoms.

Using narrower filters, perhaps semi-tone-spaced like Ellis’ constant-Q filter-bank (Ellis, 1992), yields a transformation closer to human hearing. However, the amount of data is greatly increased, and the increased number of channels makes processing more complex than in our current system. A new system tailored for multiple channels is worth exploring in future work.

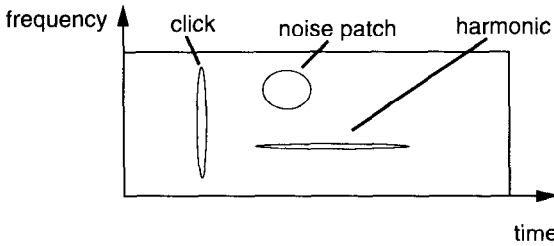


FIG. 19.4. Clicks, harmonic components and noise patches on a spectrogram.

Harmonic Components, Clicks and Noise Patches. On a spectrogram, the energy is not uniformly distributed, but tends to cluster. Grouping together energy that is adjacent in time and frequency, and parameterizing these groupings allows a great reduction of the amount of data. Fig. 19.4 shows three possible groupings of energy adjacent in time and frequency.

Musical instruments and speech show tracks on their narrow-band spectrograms, that is, lines of one frequency with a continuity in time. Smith and Serra (1987) describe a method to extract tracks from a short-time Fourier transform (STFT) spectrogram. Ellis obtains tracks from his constant-Q spectrogram (Ellis, 1992).

Tracks describe harmonic components, but are not well suited for clicks (broadband, short time) or noise patches (energy spread over both time and frequency). For those, a richer representation is required that allows atoms to span both time and frequency intervals. Matching Pursuit (Mallat and Zhang, 1992) is a method that can possibly extract such diverse atoms from sounds.

19.1.4 High Level: Distribution of Atoms

The high level of our two-level sound representation is concerned with the distribution of the sound atoms extracted at the low level. We identify periodic and stochastic (random) distributions of atoms, as well as co-occurrence and sequences of atoms. These different ways of distributing atoms are not exclusive of each other; they can be mixed and even combined in a hierarchy.

A sound in which similar atoms occur at regular interval in time is said to have a periodic distribution. Textures such as engine sounds have a periodic distribution. In a stochastic distribution, atoms occur at random times but obey some arrival rate. Rain and applause are examples of textures with a stochastic distribution of atoms. Different atoms that occur at the same time are said to co-occur. The impact of objects makes a sound where atoms co-occur in different frequency bands. Atoms also often occur in predictable sequences.

As an example, our photocopier makes a sucking sound in which many frequency components have high energy (co-occurrence). The sucking sound is followed (sequence) by a feeder sound. The suck-feed sequence is repeated

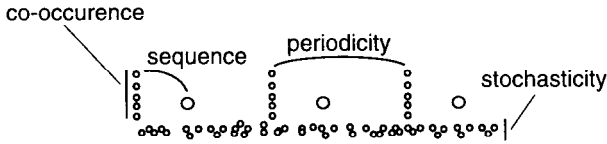


FIG. 19.5. Example of distributions of atoms: the Copier.

sequentially (periodicity). At all times there is a low rumble (stochasticity). Fig. 19.5 is a stylized representation of those four kinds of distributions.

Occurrences of atoms can be grouped into a hierarchy, for example, the sucking sound (a co-occurrence) is periodic. The high level model should address all four kinds of distributions, as well as hierarchic distributions of distributions.

Unified Description of Distribution. The method we use for characterizing the distribution of atoms (the cluster-based probability model) does not assume that the texture takes a particular distribution. Instead, it tries to characterize the distribution of atoms by keeping statistics on the most likely transitions.

19.1.5 Second Time Constraint: Attention Span

The sound of cars passing in the street illustrates an interesting problem: if there is a lot of traffic, people will say it is a texture, whereas if cars are sparse, the sound of each one is perceived as a separate event. We call *attention span* the maximum time between events before they become distinct. A few seconds is a reasonable attention span, but once again, there is no sharp boundary.

We therefore put a second time constraint on sound textures: high-level characteristics must be exposed or exemplified (in the case of stochastic distributions) within the attention span of a few seconds.

This constraint also has a good computational effect: It makes it easier to collect enough data to characterize the texture. By contrast, if a sound has a cycle of one minute, several minutes of that sound are required to collect a significant training set. This would translate into a lot of machine storage, and a lot of computation.

19.1.6 Summary of Working Definition of Sound Texture

- (1) Sound textures are formed of basic sound elements, or atoms;
- (2) atoms occur according to a higher-level pattern, which can be periodic, random, or both;
- (3) the high-level characteristics must remain the same over long time periods (which implies that there can be no complex message);

- (4) the high-level pattern must be completely exposed within a few seconds (“attention span”);
- (5) high-level randomness is also acceptable, as long as there are enough occurrences within the attention span to make a good example of the random properties.

19.2 A METHOD FOR HIGH-LEVEL CHARACTERIZATION: THE CLUSTER-BASED PROBABILITY MODEL

To characterize the high-level transitions of sound atoms, we use the cluster-based probability model (Popat and Picard, 1993). This model summarizes a high dimensionality probability mass function (PMF) by describing a set of clusters that approximate the PMF. Popat and Picard have used the cluster-based probability model for visual textures and image processing.

19.2.1 Overview

The cluster-based probability model encodes the most likely transitions of ordered features. Features (in our case sound atoms) are put in vectors, and the order within the vector reflects a pre-established order of features in time and frequency. The features used to encode the transitions are taken in the neighborhood of the current feature; therefore we call the set of the relative positions of the conditioning features a neighborhood. The vectors formed from a training sound are clustered to summarize the most likely transitions of atoms.

The input to the analyzer are a series of vectors in N-dimensional space representing the features of the training data. The vectors are clustered using a K-means algorithm (Therrien, 1989), slightly modified to iteratively split its clusters. The centroid of each cluster, its variances and relative weight then form a lower-dimensionality estimate of the statistics of the training vectors. The next section describes the use of a cluster-based probability model to characterize transitions of features, and its application to characterization of sound textures for resynthesis.

19.2.2 Application to sound texture characterization

The model attempts to characterize a sound texture, which we call the *training signal*. First the sound must be put in digital form, by sampling the analog signal. The simplest features for an audio signal are the digital values output by the analog-digital process. To study the transitions of features, we build vectors of features ordered in time but not necessarily adjacent or even equally spaced in

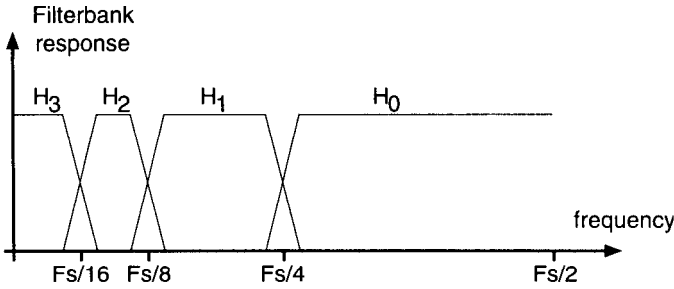


FIG. 19.6. Frequency responses for the 3-level tree-structured QMF filter bank.

time. Think of a ruler with holes punched in it; when you put the ruler over the sampled signal, you see values in each hole: This string of values is a feature vector. (Later we work with vectors spanning both time and frequency, and the 1-D ruler will become a 2-D neighborhood mask.) Different feature vectors can be obtained by sliding the ruler. If you slide the ruler over the length of the training signal, you obtain a set of training vectors. We use this set of training vectors to demonstrate the cluster-based probability model.

Each vector in the training set is an observed transition of features in time. Given that the training set is large enough to contain many observations of each likely transition, we can build a model that clusters the likely transitions and assigns a probability to each.

Each transition (training) vector is a point in a d -dimensional vector space. We assume that the signal is characterized by a probability mass function (PMF) in this vector space. Then the training vectors are samples of this PMF. Likely feature transitions correspond to dense areas of the PMF; we summarize the likely transitions by identifying clusters of points in vector space. For each cluster, we compute the center and variance of the points, and we assign a probability for the cluster based on the number of points, that is, we estimate the PMF by “histogramming” the samples of the process.

Now suppose that we have a piece of a signal, and that we want to add one likely feature at the end. To synthesize this new feature value we need to estimate its PMF conditioned on the feature values preceding it. By placing the vector-building ruler so that the last hole faces the desired new feature position, a vector identifies the desired region of space for $d-1$ dimensions. We compute the effect of each cluster along the unknown axis to get an estimate of the distribution of the value (one-dimensional PMF) at that last position. The process can be repeated to resynthesize long portions of signal.

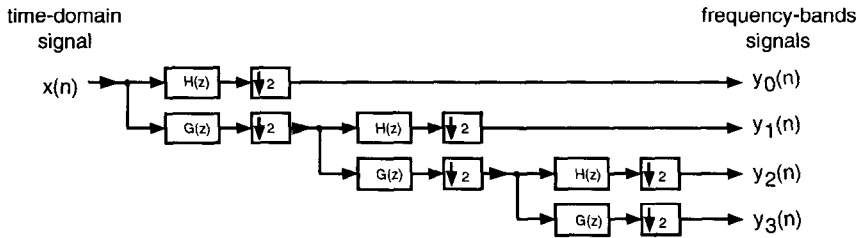


FIG. 19.7. A 3-level tree structured QMF filter bank.

19.3 RESYNTHESIS OF SOUND TEXTURES

19.3.1 Low-Level: Atom Extraction

We choose to do a frequency decomposition of the incoming sound using a binary tree structured QMF (Quadrature Mirror Filter) filter bank (Vaidyanathan, 1993). This filter bank shows a great simplicity in design. Each band (except the lowest) is one octave wide, so that the bandwidth halves at every stage, which satisfies the constant-Q criterion (Fig. 19.6). An example of a simple tree structured QMF filter bank structure is shown in Fig. 19.7.

19.3.2 High Level: Transitions of Atoms

Dimensionality Problem. Each point in the feature vector corresponds to a dimension in PMF space. In order to get a reasonable estimate of the PMF, the space must be well covered by training vectors. The more dimensions this space has, the more difficult it becomes to fill it with training vectors. We kept the number of dimensions to a maximum of 14.

Because each point in the conditioning neighborhood adds one dimension in our space (many if we use a frequency transform), we see clearly that the number of points in the neighborhood has to be kept as small as possible.

Because the transitions of atoms are not the same in all frequency channels, a different model is used for each channel in the current implementation. This multiplies the overall computational complexity, and it is another reason to keep the size of each model to a minimum.

Neighborhood Masks. To build the neighborhoods, we used only values from the past (causality), and we estimate roughly that a sample can be correlated to previous samples up to a second before.

For a given data point, the most basic neighborhood is the previous data in the same frequency band. A feature depends a lot on the values that

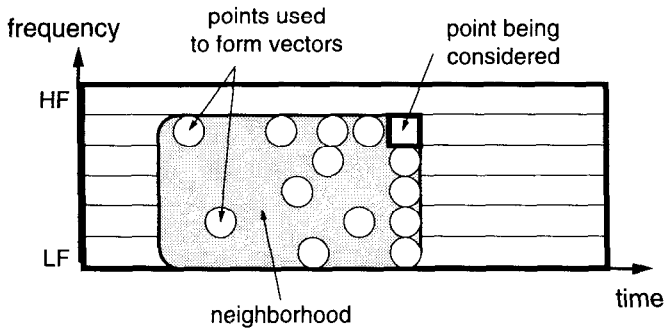


FIG. 19.8. Neighborhood on a 6-band time-frequency diagram.

immediately precede it, so some points in the close past should be part of the mask. However, to capture long-time transitions, the mask should also include some points further in the past. The general rule we use is that the points should be spaced further apart as the delay in the past increases, to keep the number of points small. In Fig. 19.8, those are the circles on the top horizontal line.

Another important part of the neighborhood is made up of the points that occur simultaneously in other frequency bands. These form the vertical line of Fig. 19.8. If the number of bands is low (as in our case), we can use a point from each band.

During resynthesis, the neighborhood should use only points already generated. Since the signal is resynthesized channel by channel starting at the low frequencies, the neighborhoods we used had atoms only in the same frequency band or lower frequency bands, as in Fig. 19.8.

19.3.3 Results

Two sines. As a reality check on the model, we tried a signal composed of two sine waves of non-integer frequency ratios as a simple test of the analysis-synthesis cycle. This signal, as all others we used, is sampled at 22.05kHz. We did not use any filtering; the features were chosen as the instantaneous sampled value of the signal. A short run of the original signal and the resynthesis are shown on Fig. 19.9.

The model has captured most of the sound; one can see that the waveform is reproduced, with some glitches. The resynthesized signal sounds like the training signal with a aliasing-sounding high-frequency hiss superposed.

The glitches could be due to an insufficient number of clusters to code all possible transitions: the signal would err for a few samples before the neighborhood would again correspond to an existing cluster.

The signal exhibits a marked start-up stage before becoming stable, as can be seen on Fig. 19.10. This is due to the absence of points to fill the neighborhood at the beginning: the model starts in a random state, and

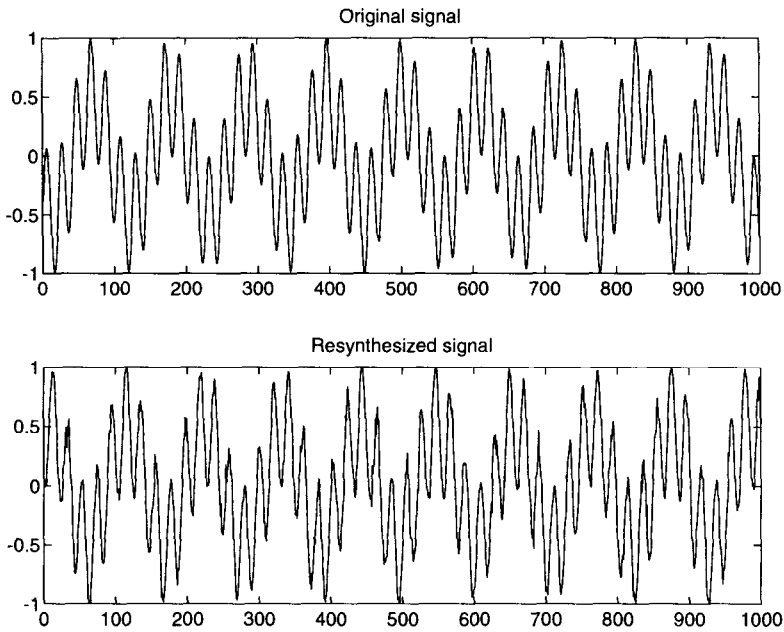


FIG. 19.9. Two sine waves: original and resynthesis.

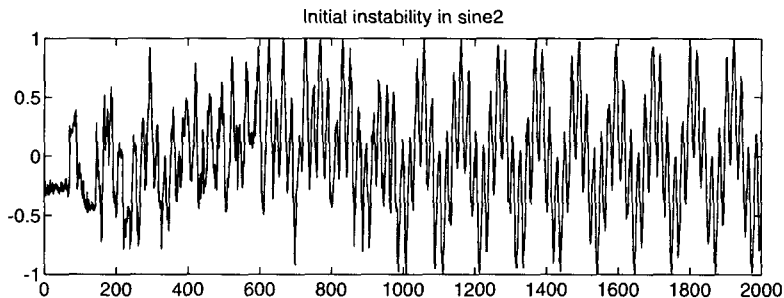


FIG. 19.10. Two sine waves: start-up stage.

eventually reaches known transitions. This problem becomes much more severe with more complex sounds, as we will see in the photocopier example.

Photocopier. Our next resynthesis attempt was done on a photocopier sound that consists of a dominant fan noise with repetitive feeder noises. The fan noise has its energy spread unequally among frequencies, like a white noise passed through a filter with resonances. The feeder sound is similar to a low frequency burst followed by a broadband click occurring at roughly equal time delays (see Fig. 19.5). Details of both the waveform and the spectrograms of the copier

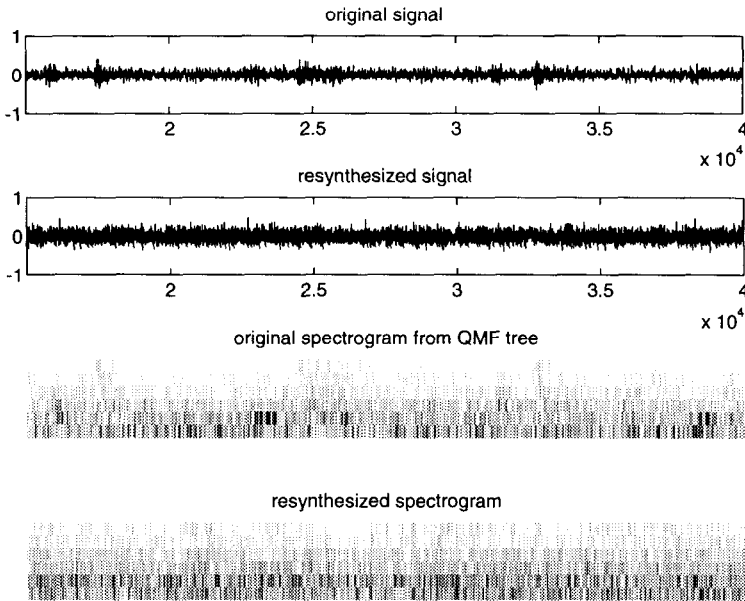


FIG. 19.11. Time signal and spectrogram of the original and resynthesized copier signal.

example can be seen in Fig. 19.11; the clicks occur around 1.7 , 2.5 and 3.3×10^4 in the time scale. (Note that the time scale is subsampled by 2, so there are 11k spectrogram slices per second).

Setting the parameters was the primary concern. For our example, the filter bank was chosen to have six frequency bands: 0-345Hz, 345-691Hz, 691Hz-1.38kHz, 1.38-2.76kHz, 2.76-5.5kHz, 5.5-11kHz (these are dependent on the sampling frequency of 22.05kHz). For each frequency band, we took about 8000 vectors (every third point) of $N=14$ points, and $M=128$ clusters. Reconstruction was done for 51200 time slices.

The neighborhoods for each frequency band chosen look very much like that in Fig. 19.8. For the lower frequency bands, the points that could not be allocated to low frequencies were used to increase the number of points in the past of the same frequency band, so that the total number of points in all neighborhood masks was constant at 13 (current point + 13 = 14 = N).

The resynthesis has kept most of the flavor (resonances) of the noise, but there are many more clicks, and they come at very irregular intervals (see Fig. 19.11). At the beginning of the resynthesis, there is no neighborhood present, so the model generates a transitory signal. We suspect that in this case, it has generated many feeder-clicks early on, and all these clicks are repeated periodically, so that there are many clicks in a time interval that should contain only one feeder sound.

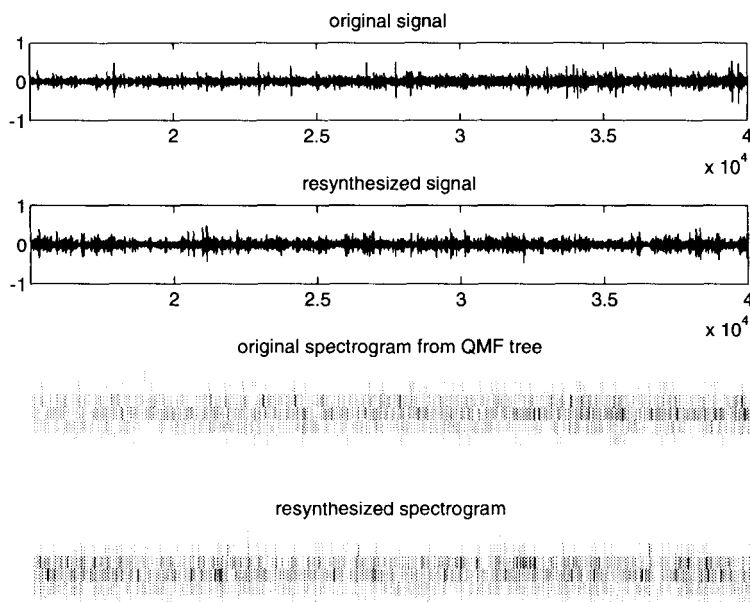


FIG. 19.12. Applause example.

Applause. A more convincing resynthesis was made from the sound of a crowd applauding. The atom extraction, model size and neighborhood masks are the same as those for the copier example. The resynthesis is a bit rough, but recognizable as applause. The spectrogram is shown on Fig. 19.12. A sound sample is available from <http://www.media.mit.edu/~nsa/IJCAI-95.html>.

Analysis of Results. The signals that were successfully resynthesized were all very stochastic in nature, with the exception of the two sines, which are deterministic. Attempts to resynthesize signals with periodicities in the order of hundreds of milliseconds, such as the sound of an helicopter or of a motorcycle, were much less successful. The resynthesis for these had a similar frequency content, and even a similar “flavor” to them as the originals, but the long-time periodicity was not properly reproduced.

All the resyntheses used a six channel filterbank. This is certainly too crude an approximation for the cochlea. Using the instantaneous amplitude in each filter channel is also a very crude feature selection. Nonetheless, the model has been able to resynthesize sound textures with some success. This leads us to believe that the model is useful and that the resynthesis could be improved by refining the atom selection and taking care of the start-up conditions.

19.4 FUTURE DIRECTIONS

19.4.1 Improvements to the Resynthesis

Increasing the dimensionality of the models used may seem like a way to model more transitions and interrelations between atoms, and thus improve analysis. However, it could be argued that if the features - atoms - being fed into the model are not good at representing the training data, then the model will not be able to summarize the transitions, however large the dimensionality of the model.

Better Atoms. In the “applause” example, the “natural” atoms are individual hand claps; is the analysis method good for characterizing this? If we were to choose individual hand claps as atoms, we would need both a “clap extractor” and a “clap resynthesizer” - which could be simply a single hand clap sample player. The resynthesis would most probably greatly improve, but the analyzer-resynthesizer combination would perform poorly on textures other than applause.

The main direction of improvement lies in choosing more meaningful atoms, and changing the atom extraction method accordingly. Somewhere between the simplistic “energy in a frequency band” atom and the specific “single hand clap” atom lies a set of physically meaningful yet general atoms. Do “harmonics, clicks and noise patches” (in section 19.1.3) form such a set?

Start-Up Condition. The initial condition is often problematic for resynthesizing sound textures: in the start-up stage, the neighborhood is mostly empty, which brings the model to very sparsely populated areas of the PMF space; the model might never get to the dense areas of the space. For those textures where the initial condition is important, it is possible to provide a seed -- a snippet of the original texture to build the first neighborhoods.

Experiments have shown that textures with strong periodic components - helicopters, motorcycles - do not resynthesize well. These textures are also very likely to be hurt by an improper start-up. Providing a sine wave of the fundamental frequency of the training texture as an initial condition might be a way to help the model converge in the right region of space for those textures.

Computation Speed. Another area requiring improvement is the computation speed. Current times on a DEC Alpha are roughly one hour of computation time for each second of resynthesized sound. Most of the machine time is spent computing the effect on the PMF of clusters that are so far from the current quadrant in the vector space that their contributions are negligible. Performance could be largely improved if those computations were avoided from the start. Preliminary results show that eliminating negligible clusters can speed up execution by a factor of 5 (Popat and Picard, 1997).

19.4.2 Classification of Sound Textures

There are many other interesting applications of the sound texture model that revolve around classification and identification. All involve comparing an unknown texture with one or more known “template” textures. Here are a few simple classification problems:

- (1) compare two textures
- (2) classify a texture as belong to one of several classes, possibly with confidence level for the result.
- (3) classify a texture as belong to one of several perceptual classes, each with multiple templates (e.g. watery sounds have templates for rain and bubbles).

All the preceding problems require a distance measure between textures. With the cluster-based probability model, we distinguish two main ways to compare two textures, resulting in two different distance measures:

- (1) compare the transitions of atoms in the unknown texture with the model for the known texture (model fit, e.g., Bayesian classification), and
- (2) compare the model extracted from the unknown texture with the model for the known texture (model comparison).

Both classification schemes assume that sound atom extraction is performed in the same way on all textures (unknown and templates), and that transition vectors are built with the same neighborhoods. Bayesian classification is straightforward and directly provides a probability that the unknown texture fits the model, but provides no insights about how the unknown and the template differ. Comparing models is much more complex because there is no obvious practical distance measure between models. However, comparing models could lead to discovery of how global parameters of texture - like periodicity - are expressed in the models.

19.5 CONCLUSION

We have presented a restricted definition of a sound texture for machine processing, which requires long-term stationarity of the characteristics of the sound. We model the textures as being composed of two levels: simple sound atoms and the distribution of the atoms. A cluster-based probability model is used to characterize the high level of sound textures. The model is then used to resynthesize textures that are in some cases similar to originals. The main source of improvement to the resynthesis is thought to be in a better selection of the type of atoms. Finally, we propose to use the model for classification of sound textures.

REFERENCES

- Ellis, D. (1992). *A Perceptual Representation of Audio*. Cambridge, Massachusetts: Master's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology.
- Mallat, S. & Zhang, Z. (1992). *Matching Pursuit with Time-Frequency Dictionaries*. New York: Courant Institute of Mathematical Sciences. (Technical Report No. 619)
- Popat, K. & Picard, R. (1993). A novel cluster-based probability model for texture synthesis, classification, and compression. Cambridge, Massachusetts: *Proceedings of SPIE Visual Communications '93*.
- Popat, K. & Picard, R. (1997). *Cluster-based probability model and its application to image and texture processing*. IEEE Transactions on Image Processing, February 1997 (to appear).
- Reddy, D. (1976). Speech recognition by machine: A review. *IEEE Proceedings*, 64, 502-531.
- Smith, J. M. & Serra, X. (1987). Parshl: an analysis/resynthesis program for non-harmonic sounds based on a sinusoidal representation. In *Proceedings of the 1987 ICMC*, p. 290ff.
- Therrien, C. (1989). *Decision, Estimation and Classification*. New York: Wiley.
- Vaidyanathan, P. (1993). *Multirate Systems and Filter Banks*. Englewood Cliffs, New Jersey: Prentice-Hall.
- Warren, W. & Verbrugge, R. (1988). Auditory perception of breaking and bouncing events. In Whitman Richards (Ed.), *Natural Computing*. Cambridge, Massachusetts: MIT Press.

NOTE

¹ Except maybe high-rate Chinese speech for someone who does not speak Chinese.