Residue-Driven Architecture for Computational Auditory Scene Analysis

Tomohiro Nakatani, Hiroshi G. Okuno, and Takeshi Kawabata

NTT Basic Research Laboratories
Nippon Telegraph and Telephone Corporation
3–1 Morinosato-Wakamiya, Atsugi, Kanagawa 243-01 JAPAN
{okuno, nakatani, kawabata}@nuesun.brl.ntt.jp

Abstract

The Residue-Driven Architecture presented here is a model of auditory stream segregation from input sounds. A subsystem to extract auditory streams by using some sound attributes is called *an agency* and the design of each agency is based on the residue-driven architecture. This architecture consists of three kinds of agents: an event-detector, a tracer-generator, and tracers. The event-detector calculates a residue by subtracting the predicted input from the actual input. When a residue exceeds a threshold value, tracer-generator generates a tracerthat extracts an auditory stream from the residue and returns a predicted input of the next time frame to the event-detector. This approach improves the performance of segregation and the resulting system can segregate a woman's voiced stream, a man's voiced stream, and a noise stream from a mixture of these sounds. Binaural segregation is also designed by the architecture.

1 Introduction — What is Computational Auditory Scene Analysis?

AI research on the understanding of sounds has a rich history dating back to the ARPA Speech Understanding Project in the 1970's. While a great deal has been learned, systems that can understand general acoustic signals (e.g. voiced speech, music and/or other sounds) from real-world environments have not been built. There are systems that understand clean speech well in relatively noiseless laboratory environments but that cannot in more realistic, noisier environments. At a crowded party, one can attend one conversation and then switch to another. This phenomenon is known as the *cocktail-party effect* and it shows that humans can selectively attend to sound from a particular source even when it is mixed with other sounds. Computers also need to decide which parts of a mixed acoustic signal are relevant to a particular purpose – which part should be interpreted as speech, for example, and which should be interpreted as a door closing, an air conditioner humming, or another person interrupting.

A number of researchers have therefore concluded that research on speech understanding and on non-speech understanding need to be unified within a general framework. One such framework is suggested by Bregman's book, *Auditory Scene Analysis* [Bregman, 1990], which discusses the psycoacoustic aspects. This work has inspired a number of attempts to model what is known about the human auditory system. It has also encouraged researchers to explore more general models of the structure of sounds in order to deal with more realistic acoustic environments. Researchers have also begun trying to understand computational auditory frameworks as parts of larger perception systems whose purpose is to give a computer integrated information about the real world. To discriminate the AI and computer science approach from the psychoacoustic approach, it has been called *Computational Auditory Scene Analysis* (hereafter, *CASA*) [Cooke *et al.*, 1993; Nakatani, Okuno, and Kawabata, 1994].

Research topics related to CASA include modeling issues, sound understandings issues, architectural issues, control issues, representational issues and applications. They also include research on how different sensors can be integrated with models of how the human's auditory apparatus works in concert with vision and other kinds of sensation. Here, we focus on system architecture based on the multi-agent paradigm. The multi-agent system was recently proposed as a new modeling technology in artificial intelligence [Brooks, 1986; Maes, 1979; Minsky, 1986; Okuno, 1993]. We take Minsky's view that an agent has a limited capability, although an agent in Distributed Artificial Intelligence is much more powerful and more like a human being than ours is.

In this paper we present a new architecture called *Residue-Driven Architecture*. The rest of this paper is organized as follows: Section 2 discusses the related works on CASA and identifies the issues. Section 3 introduces the Residue-Driven Architecture and discusses intra-agency and inter-agency interactions. Section 4 presents the design and implementation of two agencies based on harmonics and localization (the direction of sound source). Evaluations and the conclusions are respectively given in Section 5 and 6.

2 Previous Works on Computational Auditory Scene Analysis

One of goals in computational auditory scene analysis is to understand *acoustic events*, or the sources of sounds [Bregman, 1990]. An acoustic event is represented by *auditory streams* (hereafter, simply *streams*) each of which is a group of acoustic components that have consistent attributes. Since acoustic events are represented hierarchically (e.g., as an orchestra), auditory streams have also a hierarchical structure. The process that segregates auditory streams from a mixture of sounds is called *auditory stream segregation.* Since acoustic components need to be organized into auditory streams, the segregation of auditory streams requires *the exclusive allocation* of sound to a particular stream. Auditory stream segregation is performed at various levels of perception. Some streams are very simple and are extracted according to simple attributes, while others are extracted by grouping streams segregated at earlier stage of processing.

Bregman proposed two mechanisms of auditory stream segregation: *simultaneous (spectral)* grouping, followed by *sequential* grouping [Bregman, 1990]. In the simultaneous grouping, streams are extracted from a mixture of sounds; and in the sequential grouping, streams from the same acoustic event are grouped together. No algorithms of grouping for computer implementations, however, have been proposed. Nakatani et al. also showed that Bregman's approach failed in segregating man's and woman's voiced speech in their experiments [Nakatani, Okuno, and Kawabata, 1994].

In extracting acoustic attributes, some systems assume the human's auditory model of primary processing and simulate the cochlear processing [Bodden, 1993; Brown, 1992; Slaney, Naar and Lyon, 1994]. Brown and Cooke designed and implemented the system that builds various auditory maps for input sounds and integrates them to segregate speech from input sounds [Brown, 1992; Brown and Cooke, 1992]. An auditory map represents acoustic attributes such as onset, offset, AM and FM modulations, and formants. Since the integration process becomes complicated when treating a mixture of sounds in real-world environments, the blackboard architecture [Erman *et al.*, 1980] is used to simplify this integration process [Cooke *et al.*, 1993]. The algorithm building an auditory map is executed in batch in the sense that any part of the input should be available to the algorithm at any time. Batch algorithms, however, are not suitable for providing a wide variety of system responses. If the system needs a reflective response (that is, it may react immediately without deliberate consideration), it cannot wait for auditory maps to be built: some fragmentary information may be enough to decide its behavior. Additionally, it is not easy to incorporate schema-based segregation and grouping of streams into such a system, since it does not support a mechanism for extending capabilities.

To design a more flexible and expandable system, control mechanisms are needed. IPUS (*Integrated Processing and Understanding Signals*) [Lesser *et al.*, 1993] integrates signal processing and signal interpretation into the blackboard system. IPUS has a small set of front-end signal processing algorithms (SPAs). It chooses the correct parameter setting for each SPA and computes the correct interpretation by dynamic SPA reconfiguration. IPUS views this reconfiguration as a diagnosis of discrepancy between top-down search for SPA parameter settings and bottom-up search for the correct interpretation. IPUS has various interpretation knowledge sources which understand actual sounds such as hair driers, footsteps, telephone rings, fire alarms, and waterfalls. IPUS may have problems in scaling up, because when the number of SPAs increases it may fail to choose the correct parameter settings. And to support a reflective response, another system may be needed to compute required information.

Nakatani et al. took a multi-agent approach to auditory stream segregation [Nakatani, Okuno, and Kawabata, 1994]. The HBSS (*Harmonic-Based Stream Segregation*) system was designed and developed using a multi-agent system. It uses the Fourier transformation instead of the auditory model because it is easy to implement and its properties are well known. Since the HBSS uses only harmonics as a cue of segregation and retains only information of the previous time frame, it extracts auditory streams incrementally. Although its mechanism is simple, it can segregate two streams from a mixture of man's and woman's voiced speech. The HBSS can in principle segregate any number of harmonic sounds that have no fundamental frequencies in common, but the HBSS fails in scaling up the segregation because of its imperfect exclusive sound allocation caused by its poor prediction of next inputs and because of its crude mechanism for checking the consistency of streams. The main cause of these problems is that each agent does not use any temporal information about streams. Spatial information may be also used to cope with the problems. Another error is due to background noise. The definition of noise is relative, because a noise is simply something that cannot be classified by the focused attributes.

In this paper we present a new architecture to cope with the following problems of the HBSS:

(1) imperfect exclusive allocation

(2) usage of temporal and spatial information

(3) noise treatment.

3 Residue-Driven Architecture

Auditory stream segregation systems must (1) determine that streams appear, (2) trace the streams, (3) determine that the streams have ended, and (4) resolve interference between simultaneous streams. The *Residue-Driven Architecture* (Figure 1) consists of subsystems comprising three kinds of agents: an event-detector, a tracer-generator, and tracers. A subsystem extracting auditory streams by using some auditory attributes is called *an agency.* An agency based on the Residue-Driven Architecture extracts streams as follows:

(1) An event-detector subtracts a set of predicted inputs from the inputs and sends the residue to the tracer-generator and tracers.

(2) if the residue exceeds some threshold value, a tracer-generator searches for the values of focused auditory

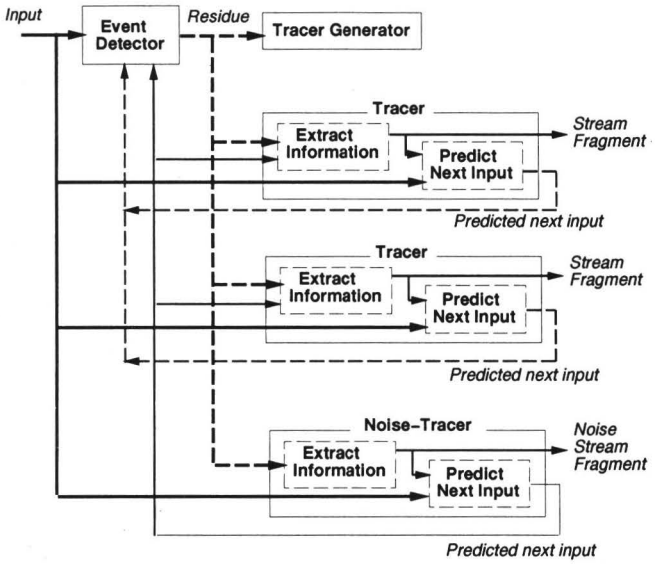


Figure 1: **Residue-driven architecture** consists of subsystems made up of three kinds of agents, an event-detector, a tracer-generator, and Tracers. This subsystem is called an *agency*.

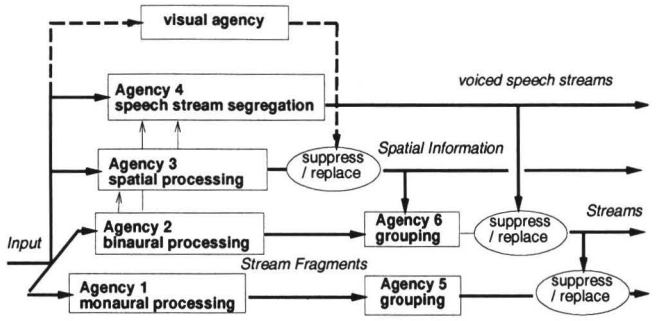


Figure 2: **Inter-agency interactions** occur in various ways. One way is *via input/output relation.* Agency 4, tracing voiced speech streams, gets information directly from agencies 2 and 3. Extracted streams can be given to the grouping agency as inputs. Another way of interaction is modeled by *the subsumption architecture*, and extracted streams may be refined by using *replace/suppress* functions. Agency 4 may substitute its extracted streams for the streams extracted by agency 5, or it may simply suppressed them.

attributes. If it finds appropriate values of the attributes, it generates a tracer to trace on the attributes. If it does not find such values, it generates a noise-tracer.

(3) Each tracer extracts a stream fragment by tracing the attributes of the stream. It also generates a predicted next input by adjusting the segregated stream fragment to the next input and sends this predicted input to the event-detector.

Once a tracer-generator generates a new tracer that starts extracting a stream, the agency returns to a stable state because the residue becomes zero unless an input does not contain a new sound. When a new sound comes in, the residue becomes nonzero and a new tracer to extract the new sound is generated and the system returns to a stable state. If a tracer predicts that the next input will be zero and the actual input is zero, the tracer terminates by itself. If the tracer-generator fails to find an appropriate attribute, it considers that a noise comes in. Since the agency treats unknown sounds as noise, the definition of noise is relative to each agency. There is only one event-detector and one tracer-generator and one noise-tracer, but the number of tracers changes according to the input.

A tracer extracts information and generates a *stream fragment.* Stream fragments are grouped into a stream. The important constraint in grouping is *exclusive allocation*, which means that each input fragment should be allocated to only one auditory stream.

Interactions between agents can be classified as intra-agency interactions and inter-agency interactions. Intra-agency interactions are performed between agents within the same agency and the main way these kinds of interactions occur is via the predicted next input. Since tracers are of the same kind, predicted next input is of the same kind and it is easy to calculate a residue by using these predicted next inputs. A noise-tracer, however, differs from other tracers and thus its predicted next input should be given to each tracer and event-detector (Figure 1). Another way of such an interaction is via shared variables.

One way inter-agency interactions occur is via the input/output relation. To model binaural hearing, for example, a pair of agencies behave like a pair of ears and each agency extracts stream fragments with spatial information. Such stream fragments are given to a grouping agency, which constructs auditory streams according to spatial information. Another way of intra-agency interaction is modeled by the subsumption architecture [Brooks, 1986], a simple example of which is shown in Figure 2.

Agency 1 in this figure extracts streams from monaural inputs, and agency 2 extracts auditory stream fragments from binaural inputs. Agency 3 extracts spatial information such as the direction of sound sources from binaural inputs. It also uses the information given by agency 2. Agencies 5 and 6 construct streams by grouping stream fragments generated by agencies 1 and 2. Agency 4 extracts voiced speech stream fragments by using information extracted by agencies 2 and 3. Agency 4 may replace the output of agency 6 with its own output or simply suppresses the output. Streams extracted by monaural processing may be replaced by those of binaural processing. The output of agency 3 may be replaced by a visual agency (e.g., a video-camera tracking system). The reason that the grouping agency is separated from agency 1 or 2 is that it can be modeled by the Residue-Driven Architecture [Nakatani et al, 1995b].

4 Designing Agencies

In designing an agency, the three kinds of agents should be specified. In this paper we describe the design and implementation of a harmonic segregating agency and the design of a harmonics-based localizing agency.

4.1 Harmonics Segregating Agency

Before describing each agent, we define several terms [Nakatani, Kawabata, and Okuno, 1995a]. First, the *harmonic intensity* $E_t(\omega)$ of the sound wave $x_t(\tau)$ at time frame t is defined as follows:

$$E_t(\omega) = \sum_k \| H_{t,k}(\omega) \|^2,$$

$$\text{where} \quad H_{t,k}(\omega) = \sum_\tau x_t(\tau) \cdot \exp(-jk\omega\tau).$$

Here, τ is time, k is the index of harmonics, $x_t(\tau)$ is the residual input, and $H_{t,k}(\omega)$ is the sound component of the kth overtone. Since some sound components, in particular for larger k's, are usually destroyed by interfering sounds, the simple $H_{t,k}(\omega)$ cannot be used. To improve the sound consistency check, a *valid overtone* for a harmonic stream is defined. An overtone is *valid* if the intensity of the overtone exceeds a threshold (Equation (1)) and the local time transition of the intensity can be approximated in a linear fashion (Equation (2)):

$$\| H_{t,k}(\omega) \| > c \cdot \| \mathrm{DFT}_t(k \cdot \omega) \|, \tag{1}$$

$$\sigma_{t,k}^2 < p \cdot M_{t,k}. \tag{2}$$

Here, $\mathrm{DFT}_t(\omega)$ is the frequency component of the input at frequency ω and at time frame t. In calculating the value of $M_{t,k}$ and $\sigma_{t,k}$, we use temporal information about fundamental frequencies. Let $\bar{\omega}_{\tau,k}$ be an average frequency of the kth overtone over m consecutive time frames at time τ. And $\bar{H}_{n,k}(\bar{\omega}_{n,k})$ is defined as $\| \mathrm{DFT}_n(k \cdot \bar{\omega}_{n,k}) \|$. $M_{t,k}$ is the mean value of $\| \bar{H}_{n,k}(\bar{\omega}_{n,k}) \|$ over m time frames. $\sigma_{t,k}^2$ is the variance of $\| \bar{H}_{n+1,k}(\bar{\omega}_{n+1,k}) - \bar{H}_{n,k}(\bar{\omega}_{n,k}) \|$ over m time frames.

The *valid harmonic intensity* $E'_t(\omega)$ is defined as the sum of the $\| H_{t,k}(\omega) \|$ over valid overtones. A *dormant period* for a noise stream is also defined. The period is *dormant* if there are only nonharmonic sounds. The average spectral intensity of the noise is calculated during the dormant period.

Tracer-Generator and Event-Detector

The tracer-generator is designed as a set of pitch watcher agents. The set of pitch watcher agents behave like a simple filter bank, but not auditory filter bank. Each pitch watcher has its own frequency region (about 25 Hz wide) and watches a residual input to detect the appearance of a new harmonic stream whose fundamental frequency is in its region. A pitch watcher is activated when the following conditions are satisfied: (a) $E'_t(\omega)/E_t(\omega) > r$ and (b) there is a power peak near frequency ω in the residual input, where ω is the frequency that maximizes $E_t(\omega)$ within the region. Since there is more than one activated pitch watcher, the pitch watcher that gives the maximum $E_t(\omega)$ is selected by the tracer-generator and generates a tracer. If there is no active pitch watcher during a dormant period, the noise-tracer is activated. The current setting of constants is as follows: $c = 0.15, p = 0.05, r = 0.1$, and $m = 10$.

Harmonics tracer

A harmonics tracer gets the initial fundamental frequency from a pitch watcher when it is generated. At each residual input, each harmonics tracer extracts the fundamental frequency that maximizes the valid harmonic intensity $E'_t(\omega)$. It then calculates the intensity and the phase of each overtone by evaluating the absolute value and the phase of $H_{t,k}(\omega)$. It generates a predicted next input in a waveform by adjusting the phase of its overtones to the phase of the next input frame. The event-detector calculates a residue by subtracting the predicted inputs from the actual input [Nakatani, Okuno, and Kawabata, 1994; Ramalingam and Kumaresan, 1994]. Each tracer recovers its input by adding its predicted input to the residual input before calculating the fundamental frequency. If there are no longer valid overtones, or if the intensity of the fundamental overtone drops below a threshold value, it terminates itself.

Noise tracer

The noise tracer segregates the static noise stream according to the average spectral intensity [Boll, 1979]. It calculates the spectral intensity time average of the residual input during the dormant period. The noise tracer sends a predicted next input to other agents by sending the spectral intensity. When a tracer receives a spectral intensity, it estimates the intensities of its sound components at each frequency by subtracting the predicted values. The predicted next input of the noise tracer inhibits the generator from generating unnecessary tracers and makes harmonics tracers robust against a nonharmonic noise. The noise tracer calculates average spectral intensity for a long-time range as well as for a short-time range, and it terminates itself when the short-time range average intensity drops below a threshold value.

4.2 Harmonics-based Localizing Agency

The harmonic agency uses monaural (single-channel) input. If multi-channel inputs from a pair of microphones or a microphone array is available, localization, or the direction of a sound source, can be also used to segregate auditory streams. In fact, binaural processing of signals or spatial hearing is said to play a critical role in the cocktail-party effect [Blauert, 1983]. There are several ways to extract spatial information from binaural input. One common way is called *coincidence model*, which calculates the interaural difference in time that the same sound arrives at each of a pair of microphones [Jeffress, 1948]. Another is to use the interaural intensity difference. These two information can be extracted by calculating interaural cross-correlation, which is based on the auditory model. Bodden used both information to get spatial information and to control the parameters of filters to extract one sound from a mixture of sounds [Bodden, 1993]. Some research also uses microphone array systems [Stadlerand Rabinowitz, 1993].

Figure 3: The structure of the binaural tracer-generator. Interaural coordinator orders a pair of tracer-generator to generate a tracer to extract the same stream if it detects a new sound on either side or both sides.

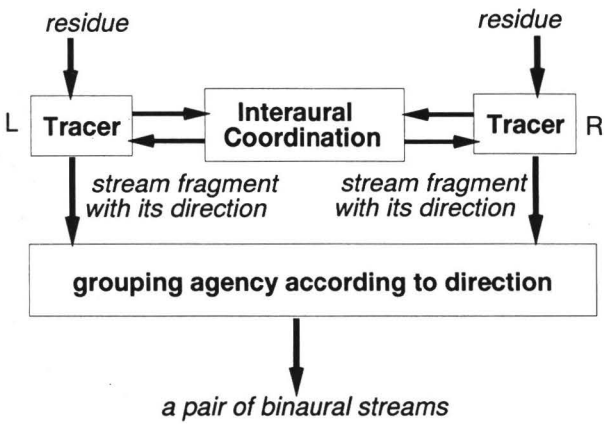


Figure 4: The structure of the binaural tracer and grouping agency. Interaural coordinator agent determines the parameters, such as fundamental frequency and direction, of the stream being extracted by a pair of tracers.

Since the design of harmonics segregating agency is independent of the (human) auditory model, we take the same approach to design a localizing agency. The localizing agency consists of two agencies: harmonics-based binaural segregating agency and binaural grouping agency [Nakatani et al, 1995b]. Harmonic-based binaural segregating agency is modeled by the Residue-Driven Architecture and is an extension of harmonics segregating agency described in the previous subsection. Its event-detector is the same as that of monaural system. The structure of its tracer-generator and tracers are shown respectively in Figure 3 and 4. The binaural tracer-generator consists of a pair of tracer-generators and an interaural coordinator. Its interaural coordinator takes candidates of new sounds from a pair of tracer-generator and orders them to generate a binaural tracer to extract a stream.

A binaural tracer consists of a pair of tracer and an interaural coordinator. Its interaural coordinator takes information about harmonic structure from a pair of tracers and determines the fundamental frequency and direction of the stream being traced by calculating the interaural difference in time and the interaural intensity difference. A pair of tracers extract stream fragments with their direction, which are organized into streams by grouping according to their directions. This grouping

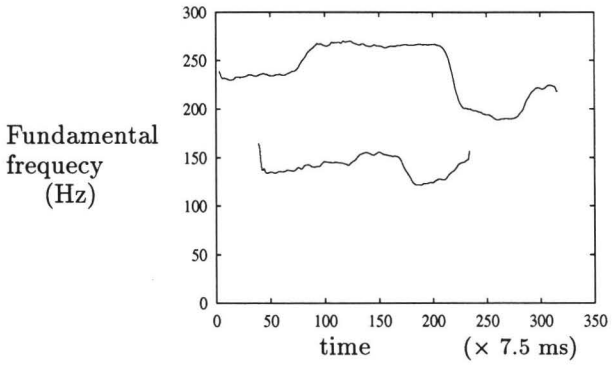


Figure 5: Fundamental frequency patterns of the benchmark sounds: man's voiced speech (lower curve) and woman's voiced speech (upper one).

Table 1: Benchmark mixture of sounds

Benchmark mixture	Man's voice	Woman's voice	white noise
1	0 dB	-1.6 dB	—
2	0 dB	-1.6 dB	10 dB
3	0 dB	-1.6 dB	5 dB
4	0 dB	-1.6 dB	0 dB
5	0 dB	-1.6 dB	-2.8 dB

agency is also modeled by the Residue-Driven Architecture. In this case, the directional information is used to generate a tracer which constructs a pair of binaural streams by grouping stream fragments of the same direction.

The merit of using harmonics is that it is easy to calculate the two kinds of interaural differences. Otherwise, we have to use spectrum for frequencies up to about 1.5 KHz to calculate them and use the envelop of sounds instead for frequencies of more than 1.5 KHz.

5 Evaluations

5.1 Evaluations of Harmonics-based Segregation Agency

We evaluated the performance of the system by using a mixture of a man's voiced speech and a woman's voiced speech, both saying "a-i-u-e-o". Figure 5 shows the fundamental frequency patterns of two speeches. The upper curve is that of the man's voiced speech and the lower one is of the woman's voiced speech. There is no common fundamental frequency, but there are several common overtones. We also used other four sets of mixed sounds by adding different power levels of white noise to it (see Table 1). Sounds are put into the system at each time frame (30-ms frame period, with a hamming window).

Experiment 1

The first experiment compared the proposed system based on the Residue-Driven Architecture and the HBSS system. Figure 6 shows fundamental frequency patterns

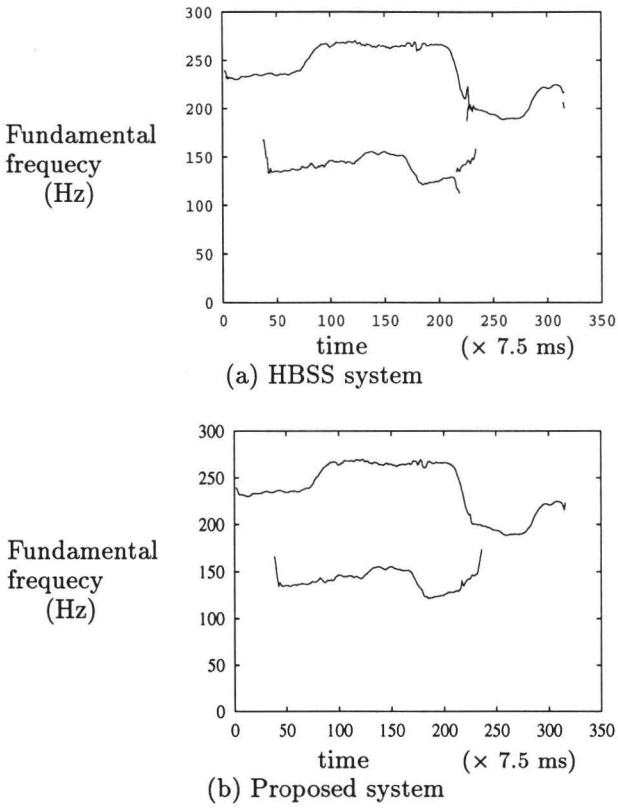


Figure 6: **Experiment 1**. Comparison of fundamental frequency patterns of streams segregated from benchmark mixture 1 (a) by the HBSS system ([Nakatani, Okuno, and Kawabata, 1994]) and (b) by the proposed system based on Residue-Driven Architecture.

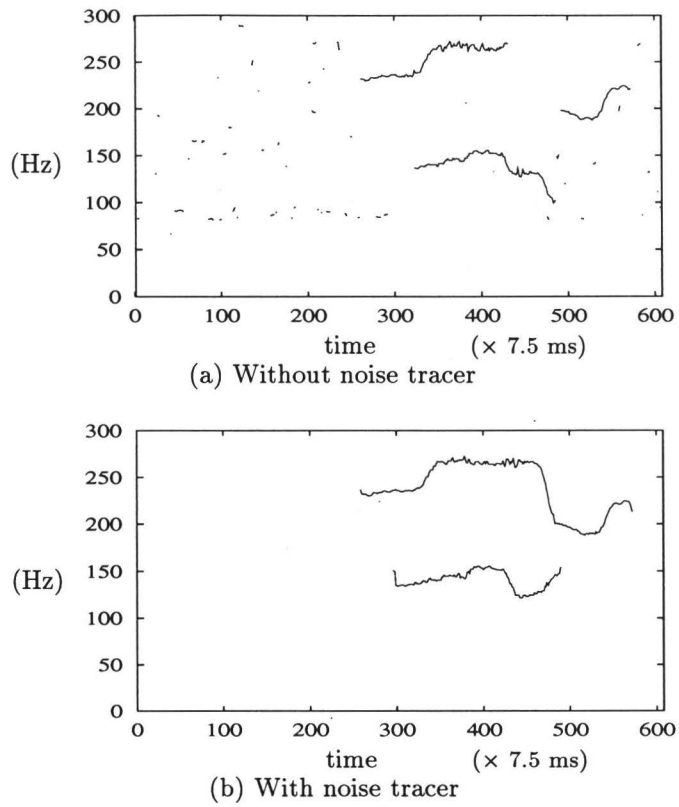


Figure 7: **Experiment 2**. Effect of noise tracer in segregation of benchmark mixture 4, where the power of white noise was the same as that of the man's voiced speech.

of streams segregated from benchmark mixture 1 by each system. Only two harmonic tracers were generated in the proposed system and thus no grouping was needed. In the HBSS system, on the other hand, 37 harmonic tracers were generated. In Figure 6(a), a woman's voiced speech was segregated as one stream, while a man's voiced speech was segregated as two consecutive streams. The segregation is much improved by using temporal information in the proposed system.

Experiment 2

The second experiment evaluated the noise tracer by benchmark mixture 4 in which the power of white noise was the same as that of the man's speech. More precisely, the benchmark first contained only white noise and then a woman started to speak "a" and was followed by a man starting to speak. Without the noise tracer, many tracers were generated in trying to find a harmonic structure in white noise, and the woman's voiced speech could not be segregated well. The quality of segregated streams of the man's voiced speech was also poor. With the noise tracer, the man's and the woman's voiced speeches were well segregated, although several false streams were also segregated. False harmonic tracers were generated but terminated immediately. The total numbers of harmonic tracers generated with and without the noise tracer were respectively 13 and 55. In the HBSS system, the total number of harmonic tracers generated with and without the noise tracer were respectively 46 and 268. The proposed system reduces the number of harmonic tracers effectively, demonstrating that sound components are allocated exclusively.

Experiment 3

The results of the third experiment, evaluating the quality of segregated sound streams with regard to spectral distortion and pitch error, are shown in Tables 2 and 3. Spectral distortion is a square root of errors of the envelop of sounds and calculated in kepstrum distance. In evaluating spectral distortion, benchmark mixture 1 to 4 were used. Their signal-noise (SN) ratios of white noise to the man's voiced speech varied from 10 dB, 5 dB, 0 dB to -2.8 dB. With noise tracer, the spectral distortion for each segregated sound was reduced by less than half.

Pitch errors, or errors of fundamental frequency, of segregated sounds were evaluated by using all benchmark mixtures in Table 1. When white noise was very small like benchmark mixture 2, the segregated man's voiced speeches were better than those segregated from benchmark mixture 1 (without noise). Pitch error without the noise tracer was small when the noise level was low, but error increased as the noise level became higher. These experimental evaluations showed that the noise

Table 2: **Experiment 3 (a)**. Spectral distortion in kepstrum distance by comparing the envelop of mixed sounds or segregated sound with that of original sound.

Benchmark	compared to	man's original voice	woman's orignal voice
1	mixed sounds	15.8 dB	12.7 dB
	segregated sound	8.7 dB	7.49 dB
2	mixed sound	21.2 dB	23.9 dB
	without noise tracer	13.9 dB	13.1 dB
	with noise tracer	11.7 dB	10.1 dB
3	mixed sound	31.5 dB	36.0 dB
	without noise tracer	23.2 dB	16.6 dB
	with noise tracer	15.5 dB	14.3 dB
4	mixed sound	45.9 dB	51.8 dB
	without noise tracer	33.3 dB	27.7 dB
	with noise tracer	23.5 dB	20.2 dB

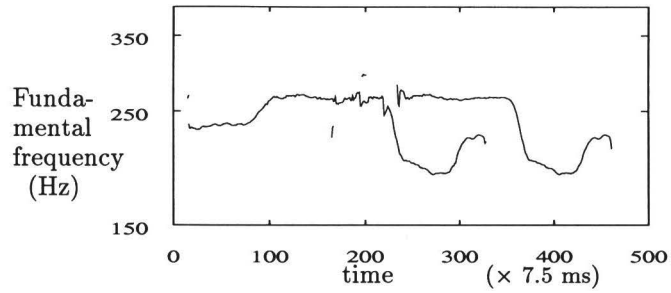
Table 3: **Experiment 3 (b)**. Comparison of fundamental frequencies between segregated and original sounds.

Benchmark		man's voice	woman's voice
1	segregated sound	0.71 Hz	0.31 Hz
2	without noise tracer	0.69 Hz	0.40 Hz
	with noise tracer	0.65 Hz	0.41 Hz
3	without noise tracer	0.82 Hz	0.51 Hz
	with noise tracer	0.80 Hz	0.55 Hz
4	without noise tracer	1.28 Hz	0.92 Hz
	with noise tracer	1.04 Hz	0.79 Hz
5	without noise tracer	4.76 Hz	1.79 Hz
	with noise tracer	1.43 Hz	1.50 Hz

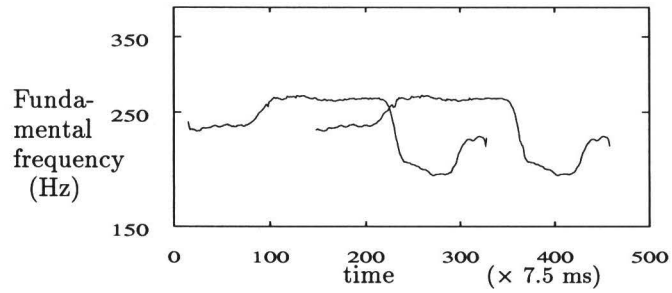
tracer is effective in improving the quality of segregated streams.

5.2 Evaluation of Localizing Agency

Localizing agency was evaluated by using a mixture of the same woman's speech saying "a-i-u-e-o" in Figure 5, synthesized by adding the first speech and the second speech starting 1.1 seconds after the first one. One speaker was positioned at -45 degree and the other was at 45 degree in the frontal plane. The proposed system with monaural input could not segregate two streams well as is shown in Figure 8(a). In particular, the initial part of second woman's voice could not be segregated, because the harmonic structures of both sounds resemble each other. The binaural system segregated two streams well as is shown in Figure 8(b), since it could use directional information to remove ambiguities of the harmonic structure between both sounds. The results of other benchmarks of different spatial settings also showed the good performance similar to Figure 8(b).



(a) Monaural system



(b) Binaural system

Figure 8: Effects of binaural system in segregating two woman's voices positined at -45 degree and +45 degree.

6 Conclusions and Future Works

This paper described the Residue-Driven Architecture for segregating auditory streams in computational auditory scene analysis. The previous HBSS system has several problems concerning imperfect exclusive allocation, usage of temporal and spatial information, and noise treatment. The Residue-Driven Architecture can easily incorporate mechanisms to cope with these problems. This architecture is used to define an agency that segregates auditory streams by tracing sound attributes. Two agencies, harmonic segregating agency and localizing agency, are presented. Both harmonic agency with the noise tracer and localizing agency improve the quality of segregation.

A lot of issues remain, since auditory stream segregation is a primitive function for computational auditory scene analysis. Okuno et al. proposed two essential problems [Okuno, Nakatani, and Kawabata, 1995].

(1) *the cocktail-party effect* — selectively attending one conversation or sound source and then changing the focus of attention to another [Okuno, Nakatani, and Kawabata, 1995].

(2) *the Prince Shotoku effect* — listening to several things at the same time [Cooke *et al.*, 1993]. This effect is named for Prince Shotoku (574–622) in Japan, who is said to have been able to listen to seven people petitioning him at the same time.

These problems require speech stream segregation, whose main issues are handling consonants or jumping sounds. There may be many clues in speech stream segregation, such as temporal structure, spectral structure, spatial structure and attributes of voiced speech.

From the view-point of AI research, the representation of voiced speech including vowels and consonants is mandatory, but as far as we know, no such representation has been proposed. We are instead using the localizing agency to extract speech streams from binaural inputs. The cocktail-party effect is seldom observed when one ear is plugged or hearing is impaired, and this is because the ability to localize a sound source is damaged. Speech stream segregation has many potential applications and we think that CASA will contribute various aspects of social life.

Acknowledgments

We thank David F. Rosenthal for cooperation in writing the introduction section of an earlier draft of the paper as well as for discussions on computational auditory scene analysis. We also thank Makio Kashino, Kunio Kashino, Masataka Goto, Rikio Onai and Ikuo Takeuchi for such discussions. We also appreciate the pointer to the literature on binaural processing given by an anonymous reviewer.

References

[Blauert, 1983] J. Blauert. *Spatial Hearing: the psychophysics of human sound localization.* MIT Press.

[Bodden, 1993] M. Bodden. Modeling human sound-source localization and the cocktail-party-effect. *Acta Acustica*, 1:43–55, Feb./Apr. 1993.

[Boll, 1979] S.F. Boll. A Spectral Subtraction Algorithm for Suppression of Acoustic Noise in Speech. In *Proceegings of the International Conference on Acoustics, Speech, and Signal Processing*, pages 200–203, IEEE.

[Bregman, 1990] A.S. Bregman. *Auditory Scene Analysis – the perceptual organization of sound.* MIT Press.

[Brooks, 1986] R.A. Brooks. A Robust Layered Control System for a Mobile Robot. *IEEE Journal of Robotics and Automation*, RA-2(1):14–23, 1986.

[Brown, 1992] G.J. Brown. Computational auditory scene analysis: A representational approach. *PhD thesis*, Dept. of Computer Science, University of Sheffield.

[Brown and Cooke, 1992] G.J. Brown and M.P. Cooke. A computational model of auditory scene analysis. In *Proceedings of the International Conference on Spoken Language Processing*, pages 523–526, IEEE, 1992.

[Cooke *et al.*, 1993] M.P. Cooke, G.J. Brown, M. Crawford and Green: Computational Auditory Scene Analysis: listening to several things at once. *Endeavour*, 17(4):186–190, 1993.

[Erman *et al.*, 1980] L. Erman, F. Hayes-Roth, V. Lesser, and R. Reddy. The Hearsay-II speech-understanding system: Integrating knowledge to resolve uncertainty. *Computing Surveys*, 12:213–253, ACM, 1980.

[Jeffress, 1948] L.A. Jeffress: A place theory of sound localization. *Journal of Comp. Physiol. Psychol.*, **61**:468–486, 1948.

[Lesser *et al.*, 1993] V. Lesser, S.H. Nawab, I. Gallastegi, and F. Klassner. IPUS: An Architecture for Integrated Signal Processing and Signal Interpretation in Complex Environments. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 249–255, AAAI, 1993.

[Maes, 1979] P. Maes editor. *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back.* special issue of *Robot and Autonomous Systems*, MIT Press/Elsevier, 1991.

[Minsky, 1986] M. Minsky. *Society of Minds.* Simon & Schuster, Inc., 1986.

[Nakatani, Okuno, and Kawabata, 1994] T. Nakatani, H. G. Okuno, and T. Kawabata. Auditory Stream Segregation in Auditory Scene Analysis with a Multi-Agent System. In *Proceedings of the Twelve National Conference on Artificial Intelligence*, pages 100–107, AAAI, August 1994.

[Nakatani, Kawabata, and Okuno, 1995a] T. Nakatani, T. Kawabata, and H.G. Okuno: A computational model of sound stream segregation with the multi-agent paradigm. In *Proceedings of 1995 International Conference on Acoustics, Speech and Signal Processing, to appear*, IEEE, Detroit, May 1995.

[Nakatani et al, 1995b] T. Nakatani, M. Goto, I. Takatoshi, and H.G. Okuno: Multi-Agent Based Binaural Sound Stream Segregation. In *Proceedings of IJCAI-95 Workshop on Computational Auditory Scene Analysis, to appear*, Aug. 1995.

[Okuno, 1993] H.G. Okuno. Cognition Model with Multi-Agent System (*in Japanese*). pages 213–225. In T. Ishida, editor. *Multi-Agent and Cooperative Computation II*, Tokyo, Japan: Kindai-Kagaku-sha, 1993.

[Okuno, Nakatani, and Kawabata, 1995] H.G. Okuno, T. Nakatani, and T. Kawabata. Cocktail-Party Effect with Computational Auditory Scene Analysis — Preliminary Report —. In *Proceedings of the Sixth International Conference on Human-Computer Interaction, to appear*, Elsevier Science B.V., July 1995.

[Ramalingam and Kumaresan, 1994] C.S. Ramalingam and R. Kumaresan. Voiced-speech analysis based on the residual interfering signal canceler (RISC) algorithm. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Vol.I, pages 473–476, IEEE, 1994.

[Slaney, Naar and Lyon, 1994] M. Slaney, D. Naar, and R.F. Lyon. Auditory Model Inversion For Sound Separation. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Vol.II, pages 77–80, IEEE, 1994.

[Stadlerand Rabinowitz, 1993] R.W. Stadler and W.M. Rabinowitz. On the potential of fixed arrays for hearing aids. *Journal of Acoustic Society of America*, **94**(3) Pt.1:1332–1342, Sept. 1993.