IPUS: An Architecture for Integrated Signal Processing and Signal Interpretation in Complex Environments*

Victor Lesser Hamid Nawab† Izaskun Gallastegi Frank Klassner

Computer Science Department
University of Massachusetts
Amherst, MA 01003
{lesser izaskun klassner}@cs.umass.edu

†ECS Department
Boston University
Boston, MA 02125
hamid@buengc.bu.edu

Abstract

This paper presents the IPUS (*Integrated Processing and Understanding of Signals*) architecture to address the traditional perceptual paradigm's shortcomings in complex environments. It has two premises: (1) the search for correct interpretations of signal processing algorithms' (SPAs) outputs requires concurrent search for SPAs and control parameters appropriate for the environment, and (2) interaction between these search processes must be structured by a formal theory of how inappropriate SPA usage can distort SPA output. We describe IPUS's key components (discrepancy detection, diagnosis, reprocessing, and differential diagnosis) and their instantiation in an acoustic interpretation system. This application, along with another in the radar domain, supports our claim that the IPUS paradigm is feasible and generic.

Introduction

In traditional knowledge-based perceptual systems [8, 17], numeric signal processing is fixed, and interpretation processes are limited to analyzing the single view afforded by this processing. This paradigm assumes that a small set of front-end signal processing algorithms (SPAs) with fixed parameter settings can produce adequate evidence for deriving plausible interpretations under all scenarios. The complex environments that next-generation systems will monitor, however, have variable signal to noise ratios, unpredictable source behaviors, and many sources whose signatures can mask or otherwise distort each other. Under the traditional paradigm, such environments often require combinatorially explosive SPA sets with multiple parameter settings to capture the variety of signals adequately [7] and to handle the variety of processing goals the current environment may dictate. To avoid this problem, we argue that knowledge-based perceptual research needs to consider a paradigm incorporating *dynamic SPA reconfiguration.* This term refers not only to reconfiguration for tracking changes in signal behavior, but also to (repeated) reconfiguration for analyzing cached data to reduce uncertainty in signal interpretations.

Research in active vision and robotics has recognized the importance of tracking-oriented reconfiguration [19], and tends to use a control-theoretic approach for making reconfiguration decisions. It is indeed sometimes possible to reduce the reconfiguration of small sets of front-end SPAs to problems in linear control theory. In general, however, the problem of deciding when an SPA (e.g. a shape-from-X algorithm or an acoustic filter) with particular parameter settings is appropriate to a given environment may involve nonlinear control or be unsolvable with current control theory techniques.

Recent systems in other fields [4, 5, 6, 9, 11] have used symbolically-oriented architectures that permit interpretation processes to reconfigure front-end signal processing. However, as the **Related Work** section will show, their architectures have not been general enough. We have developed an architecture to permit more general interaction between signal processing and signal interpretation by explicitly representing the theory underlying front-end SPAs. The *Integrated Processing and Understanding of Signals* (IPUS) architecture has two premises for complex environments. The first is that the search for correct interpretations of numeric SPAs' outputs requires a concurrent search for SPAs and control parameters appropriate for the environment. The second premise is that the interaction between these search processes must be bidirectional and structured by a formal theory of how inappropriate parameter settings or applications of SPAs lead to specific discrepancies in SPA output.

This paper presents (1) the generic architecture, (2) the IPUS components' generic design and interaction, (3) IPUS instantiated in a sound understanding testbed, (4) related work, and (5) conclusions.

*This work was supported by the Rome Air Development Center of the Air Force Systems Command under contract F30602-91-C-0038, and by the Office of Naval Research under contract N00014-92-J-1450. The content does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

The Generic IPUS Architecture

Before describing IPUS we must first discuss SPAs, the basic means for analyzing environmental signals. When applied to a signal, an SPA instance produces correlates, which serve as evidence for hypothesizing features of objects (e.g. sounds or physical objects). An SPA instance is specified by values for a generic SPA's parameters, and these values induce capabilities or limitations with respect to the scenario being monitored. We use "SPA" to refer to SPA instances. Consider the Short-Time Fourier Transform (STFT) [15] in the acoustic domain. An STFT instance has particular values for its parameters, such as analysis window length, frequency-sampling rate, and decimation factor (consecutive analysis windows' separation). Depending on a scenario's spectral features and their time-variant nature, these parameter values increase or decrease the instance's usefulness in monitoring the scenario. Instances with large window lengths provide fine frequency resolution for scenarios containing sounds with time-invariant components, but at the cost of poor time resolution for scenarios containing sounds with time-varying components.

Figure 1a shows the generic IPUS architecture. Two types of signal interpretation hypotheses are stored on the hierarchical blackboard: current signal data's interpretations and expectations about future data's interpretations.

The design of IPUS assumes that signal data is analyzed in blocks. IPUS uses an iterative process to converge to appropriate SPAs and interpretations. The following is a summary (see **Architecture Components** and [12, 14] for more detail). For each data block, the loop starts by processing the signal with an initial SPA configuration. These SPAs are selected not only to identify and track the objects most likely to appear, but also to provide indications of when less likely or unknown objects have appeared. In the next loop step, a *discrepancy detection* process tests for discrepancies between the correlates of each SPA in the current configuration and expectations based on (1) object models, (2) the correlates of other SPAs in the configuration, and (3) application-domain signal characteristics. These comparisons may occur both after SPA output is generated and after interpretations are generated. If discrepancies are detected, a *diagnosis* process then attempts to explain them in terms of a set of distortion hypotheses. This diagnosis uses the formal theory underlying the signal processing. The loop ends with a *signal reprocessing* stage that proposes and executes a search plan to find a new front-end (i.e. a set of SPAs) to eliminate or reduce the hypothesized distortions. After the loop's completion, if there are any similarly-rated competing top-level interpretations, a *differential diagnosis* process selects and executes a reprocessing plan to detect features that will discriminate among the alternatives.

IPUS is intended to integrate the search for interpretations of SPA correlates with the search for SPA parameter values appropriate to the scenario. In complex environments we argue that these searches must interact bidirectionally under the guidance of a domain's formal signal processing theory. The dual search in the framework becomes apparent with the following observations. Each time data is reprocessed, whether for disambiguation or distortion elimination, a new state in the SPA search space is tested for how well it eliminates distortions. The measurement of distortion elimination or disambiguation assumes that the system's current state in the interpretation space matches the scenario being observed. Failure to remove a hypothesized distortion after a bounded search in the SPA space will lead to a new search in the interpretation space. This occurs because the diagnosis and reprocessing results represent attempts at justifying the assumption that the current interpretation is correct. If either diagnosis or reprocessing fails, there is a strong likelihood that the current interpretation is not correct and a new search is required in the interpretation space. Furthermore, the results of failed reprocessing can constrain the new interpretation search by eliminating from consideration objects with features that should have been found during the reprocessing.

We designed IPUS to serve as the basis of perceptual systems that can manage their interpretations' uncertainty levels. Therefore, we had to provide the architecture's control framework with a way to represent factors that affect interpretations' certainties. The control framework also had to support context-sensitive focusing on particular uncertainties in order to control engagement and interruption of the architecture's reprocessing loop.

For these reasons, IPUS uses the RESUN [3] framework to control knowledge source (KS) execution. This framework supports the view of interpretation as a process of gathering evidence to resolve hypotheses' sources of uncertainty (SOUs). It incorporates a language for representing SOUs as structures which trigger the selection of appropriate interpretation strategies. Problem-solving is driven by information in the *problem solving model*, which is a summary of the current interpretations and the SOUs associated with each one's supporting hypotheses. An incremental, reactive planner maintains control using *control plans* and *focusing heuristics*. Control plans are schemas that define the strategies and SPAs available to the system for processing and interpreting data, and for resolving interpretation uncertainties. Focusing heuristics are context-sensitive tests to select SOUs to resolve and processing strategies to pursue.

Architecture Components

This section provides detailed, yet generic, descriptions of the key architectural components. Our focus is on the three roles a domain's formal signal processing theory can play in guiding interpretation and processing in

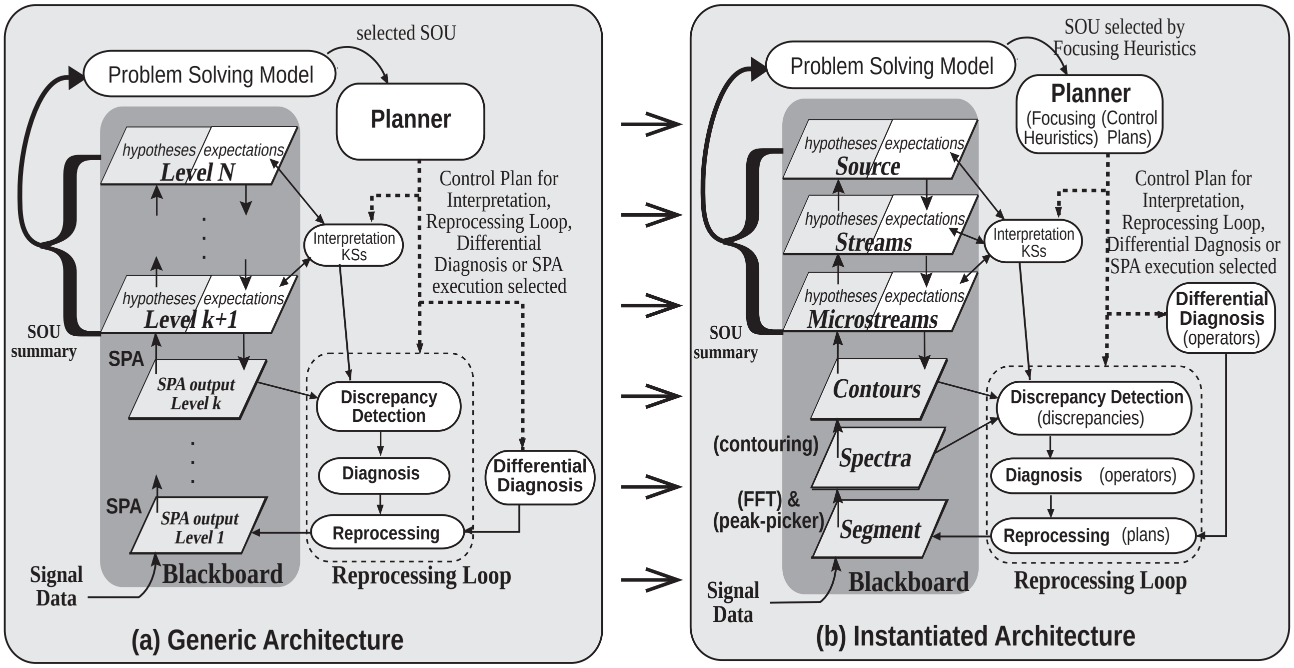

Figure 1: *1a shows the generic IPUS architecture, 1b shows the architecture instantiated for the sound understanding testbed. Solid arrows indicate dataflow relations. Dotted arrows indicate plans that the planner can pursue when trying to reduce SOUs (discrepancies) in the problem solving model that were selected by the focusing heuristics. Knowledge to instantiate the architecture for an application is shown in parentheses in 1b. Reprocessing plans can produce SPA output at any abstraction level, not just the lowest.*

a complex environment: (1) providing methods to determine discrepancies between an SPA's expected correlate set and its computed correlate set, (2) defining distortion processes that explain how discrepancies between expectations and an SPA's computed correlates result when the SPA has inappropriate values for specific parameters, and (3) specifying strategies to reprocess signals so that distortions are removed or ambiguous data is disambiguated.

We relate a signal processing theory to SPAs and their interaction with the environment using *SPA processing models*. An SPA processing model describes how the output of the SPA changes when one of its control parameters is varied while all the others are held fixed.

SPA processing models serve as the basis for defining how the parameter settings of an SPA can introduce distortions into the SPA's computed correlates. These distortions cause SPA output discrepancies. Consider an SPA processing model corresponding to the STFT's WINDOW-LENGTH parameter and how this model can be used to define distortions. Assume that an STFT with an analysis window of length W is applied to a signal sampled at rate R. If the signal came from a scenario containing two or more frequency tracks closer than R/W, Fourier theory predicts that the tracks will appear as one track in the STFT's correlates.

Discrepancy Detection

Discrepancy detection is crucial to IPUS's iterative approach. Its inclusion in the IPUS loop relies on several observations. An SPA's correlates can be compared with expectations based on object models or on *a priori* environment constraints such as maximum bounds on sounds' rate of temporal change in frequency. Most importantly, a domain's signal processing theory can specify how one SPA's correlates for a context-independent feature can serve as the basis of expectations for another SPA's output correlates. This specification can serve to check an SPA's appropriateness to the environment. It can also serve to decide where to selectively apply another SPA in the signal data stream to obtain correlates for context-dependent features. For example, in the acoustic domain a time-domain energy tracking algorithm can detect impulsive sources whose short-duration frequency components might be smoothed to indetectability in the output of an STFT with a wide analysis window. Thus, the energy algorithm can serve as a standard against which STFT output can be compared.

We categorize discrepancies in focusing heuristics for diagnostic consideration in the following order:

faults A discrepancy detected between an SPA's correlates and correlates from other SPAs applied to the same data. In [2] we discuss several fault discrepancy detection algorithms used in the sound understanding testbed. Faults are considered for diagnosis first since inconsistency among the outputs of two or more SPAs within a front-end almost always indicates a front-end's inappropriate application.

violations A discrepancy detected between an SPA's correlates and environment constraints. Violations are ranked second for diagnosis since they reflect a comparison between only one SPA's output and domain characteristics that may be incompletely specified.

conflicts A discrepancy between an SPA's correlates and the output expected based on previous high-level interpretations. Conflicts are ranked third for diagnosis since they reflect a comparison between SPA output and interpretations which may not be accurate even if they are based on appropriately-processed data.

In IPUS, conflict discrepancy detection is distributed among all KSs that interpret lower-level data as higher-level concepts. Each such KS checks if any data can support a sought-after expectation. If no such data or only partially supportive data is found, the KS records this fact as an SOU in the problem solving model, to be resolved at the discretion of the focusing heuristics. Once a data block's front-end processing is completed, a discrepancy detection KS checks if SPA correlates are consistent with each other, testing for violations and faults defined by the system designer.

An important consideration in discrepancy detection is that expectation hypotheses are sometimes only expressible qualitatively, as in the example, "During the next 400 to 800 msec, a sinusoidal component currently at 1200 Hz will shift to a frequency between 1700 and 2000 Hz." Thus, our testbed discrepancy detection components use a range calculus similar to Allen's [1] to specify discrepancies.

Discrepancy Diagnosis

The discrepancy diagnosis KS is included to take advantage of the fact that a signal domain's SPA processing models can predict the form of an SPA's correlates when the SPA's parameter values are appropriate *or* inappropriate to the current scenario.

The KS models this knowledge in a database of distortion operators. When an operator is applied to a description of undistorted SPA output, it returns the output with the operator's distortion introduced. The KS uses these operators in a means-ends analysis framework [16] to "explain" discrepancies between the expected form of an SPA's correlates and the actual form of the SPA's computed correlates. There are two inputs for this KS: an *initial state* representing the expected correlates' form and a *goal state* representing the computed correlates' form. The formal task of diagnosis is to generate an operator sequence mapping the initial state onto the goal state. Note that there is a difference between discrepancies and distortions. Distortions are used to explain discrepancies. It is also possible for several distortions to explain the same kinds of discrepancies. In the **IPUS Instantiation** section we will see how a "low frequency resolution" distortion explains 'missing' track discrepancies.

The KS's search for a distortion operator sequence is iteratively carried out using progressively more complex abstractions of the initial and goal states, until a level is reached where a sequence can be generated using no more signal information than is available at that level. Thus, the KS mimics expert diagnostic reasoning in that it offers simplest explanations first [18]. Once a sequence is found, the KS enters its *verify* phase, "drops" to the lowest abstraction level, and checks that each operator's pre- and post-conditions are met when all available state information is considered. If verification succeeds, the operator sequence and a diagnosis region indicating the hypotheses involved in the discrepancy are returned. If it fails, the KS attempts to "patch" the sequence by finding operator subsequences that eliminate the unmet conditions and inserting them in the original sequence. If no patch is possible, and no alternative explanations can be generated, the hypotheses hypotheses in the initial state are annotated with an SOU with a very negative rating.

An issue not addressed in earlier work [16] that arose in the development of IPUS is the problem of inapplicable explanations. Sometimes the first explanation offered by the KS will not enable the reprocessing mechanism to eliminate a discrepancy. In these cases, the architecture permits reactivation of the diagnostic KS with the previous explanation supplied as one that must not be returned again. To avoid repeating the search performed for the previous explanation, the KS stores with its explanations the search-tree context it was in when the explanation was produced. The KS's search for a new explanation begins from that point.

Signal Reprocessing

Once distortions have been explained, it falls to the reprocessing KS to search for appropriate SPAs and parameter values that can reduce or remove them. This component incorporates the following phases: *assessment, selection, and execution.* The reprocessing KS input includes a description of the input and output states, the distortion operator sequence hypothesized by the diagnosis KS, and a description of the discrepancies present between the input and output states. The assessment phase uses case-based reasoning constrained by signal processing theory to generate reprocessing plans that have the potential of eliminating the hypothesized distortions present in the current situation. For example, Fourier theory indicates that frequency resolution distortions, if actually present in STFT output, can be eliminated in a reapplication of the SPA with its FFT-SIZE parameter double or quadruple that of the original setting.

In the selection stage, a plan is selected from the retrieved set based on computation costs or other criteria supplied by focusing heuristics. The execution phase consists of incrementally adjusting the SPAs parameters, applying the SPAs to the portion of the signal data that is hypothesized to contain distortions, and testing for discrepancy removal. The execution phase is necessarily incremental because the situation description is at least partially qualitative, and therefore it is generally impossible to predict *a priori* exact parameter values to be used in the reprocessing.

Execution continues until the distortion causing the discrepancy is removed or plan failure occurs. Plan failure is indicated when either the plan's iterations exceed a fixed threshold or a plan iteration requires a SPA parameter to have a value outside fixed bounds. When failure occurs, the diagnosis KS can be re-invoked to find an alternative explanation for the original distortions. If no alternative explanation can be found, the hypotheses involved in the discrepancy are annotated with SOUs indicating low confidence due to irresolvable discrepancies.

Differential Diagnosis

We include the differential diagnosis KS to produce reprocessing plans that prune the interpretation search space when ambiguous data is encountered. Its input is the ambiguous data's set of alternative interpretations, and it returns the time period in the signal to be reprocessed, the evidence each interpretation requires, and the set of proposed reprocessing plans.

The KS first labels any observed evidence in the interpretation hypotheses' overlapping features as "ambiguous." It then determines the hypotheses' discriminating features (e.g., in the acoustic domain, those frequency tracks of the competing source hypotheses' models which don't overlap any other models' tracks). For each discriminating feature with no observed evidence, the KS posits an explanation for how the evidence could have gone undetected, assuming the source was present. These explanations index into a plan database, and select reprocessing plans to cause the missing evidence to appear. The KS then checks each ambiguous data region for resolution problems based on source models (e.g., a frequency region's peaks could support one source Y component or two source Z components), and selects reprocessing plans to provide finer component resolution in those regions.

The reprocessing plan set returned is the first nonempty set in the sequence: missing-evidence and ambiguous-evidence plan sets' intersection, missing-evidence plan set, ambiguous-evidence plan set. This hierarchy returns the plans most likely to prune many interpretations from further consideration. The alternative hypotheses' temporal overlap region defines the reprocessing region, and the ambiguous and missing evidence handled by the reprocessing plan set defines the support evidence. A plan from the returned set is then iteratively executed as in the reprocessing KS until either a plan-failure criterion is met or at least one support evidence element is found.

This KS's explanatory reasoning for missing evidence is primitive compared to the discrepancy diagnosis KS's. Only simple, single distortions like loss of low-energy components due to energy thresholding are considered; no multiple-distortion explanations are constructed. This design is justified because the KS's role is to quickly prune large areas of interpretation spaces, *without preference* for any particular interpretation. When a particular interpretation is preferred (rated) over alternatives and a detailed explanation for its missing support is required, IPUS control plans would instead use the discrepancy diagnosis KS, encoding the preferred interpretation in the initial state.

IPUS Instantiation

We have implemented a sound understanding testbed to test the IPUS architecture's realizability and generality (see Figure 1b). In this section we discuss one of the testbed experiment scenarios and how the architecture structured the application of acoustic signal processing knowledge to the scenario's interpretation. The discussion is not intended to illustrate specific control plans' execution or specific SOUs' generation. The testbed version described here is called configuration *C.1*. We are currently developing a second version *C.2* that still relies on the basic IPUS framework but that uses approximate-knowledge KSs to constrain the number of sound models retrieved when large sound libraries are used.

The testbed uses 1500Kb of Common Lisp code and runs on a TI Explorer II+. All SPAs are implemented in software. The testbed SPA database has 3 classes: STFT, energy tracking, and spectral peak-picking. For this experiment the source database contains 5 synthetic and noise-free real-world acoustic source models; the signal is sampled at 10KHz. The scenario and pertinent source models appear in Figure 2. The testbed was initially configured to track a hairdryer sound with two frequency components at 1000 and 1050 Hz. The configuration had a high peak-picking energy threshold to minimize the number of low-energy noise peaks produced by the hairdryer, and STFT parameter settings to provide enough resolution to separate the hairdryer's frequency components. The telephone ring and the door slam represent unexpected source events for which the testbed must temporarily switch SPA configurations if it is to identify them with sufficient certainty.

Because the testbed's SPA settings were originally set for tracking the hairdryer, the testbed must de-

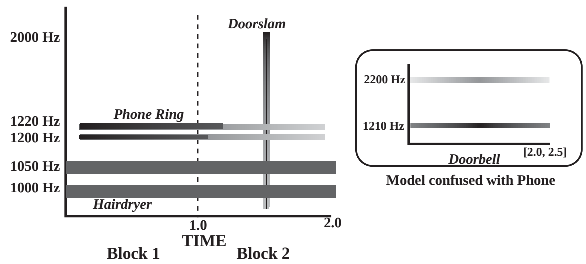

Figure 2: *Scenario and pertinent source definitions. Darker shading indicates higher signal energies.*

tect several discrepancies and perform reprocessing to reasonably analyze the scenario. In block 1's data, the $[1200, 1220]$ Hz region has insufficient resolution to display the phone ring's components due to the frequency-sampling provided by the STFT SPA's FFT-SIZE parameter value. This causes a narrow-band set of peaks with no clear energy trends to appear in the region, thus violating the noise distribution model and raising a discrepancy. The output could support the phone ring, the doorbell, or even both. Had only one candidate interpretation been identified, the testbed would have handled the violation discrepancy via the reprocessing loop. Because more than one interpretation exists, however, the testbed's focusing heuristics select differential diagnosis to resolve the interpretation uncertainty. The diagnosis finds two reasons for the confusion: the peak-picking SPA's high energy threshold designed for the hairdryer would prevent the doorbell's low-energy 2200 Hz component from appearing if it were present and the $[1200, 1220]$ region's low frequency resolution. The uncertainty is resolved in favor of the phone ring interpretation through reprocessing that doubles the FFT-SIZE and decreases the energy threshold. The phone ring's definition and block 1's interpretation generate the expectation for block 2 that it should contain the phone ring's frequencies.

Because the testbed's primary goal is to track the hairdryer, the parameter settings are reset to their original values. In block 2, the testbed detects a fault discrepancy between its time-domain energy-estimator SPA output and its STFT SPA output. The energy-estimator detects the door slam's substantial energy increase followed about 0.1 seconds later by a precipitous decrease. The STFT SPA, however, produces no significant set of peaks to account for the signal energy flux. This is because the SPA's time decimation parameter is too small. The testbed also detects a conflict discrepancy between expectations established from block 1 for the $[1200, 1220]$ frequency region and the STFT SPA's output. The STFT SPA produces a peak set with no energy trends that can support the phone ring's expected continuation because of inadequate frequency sampling in the region. Both discrepancies are resolved by reprocessing based on discrepancy diagnosis explanations. The first discrepancy is resolved through reprocessing with a larger decimation value and smaller STFT windows, while the second is resolved through reprocessing with the finer frequency sampling provided by a 2048 FFT-SIZE.

Related Work

IPUS represents the formalization and extension of concepts explored in our work on a diagnosis system that used formal signal processing theory to debug signal processing systems [16] and in work on meta-level control [10] that used a process of fault-detection, diagnosis, and replanning to choose appropriate parameters for controlling a problem-solving system.

Recent systems have begun to explore interaction between interpretation activity and signal processing. The GUARDIAN system's [9] data management component controls signal sampling rates with respect to real-time constraints. It is designed for monitoring simple signals such as heart rate and does not seem adequate for monitoring signals with complex structures that must be modeled over time. The framework is typical of systems whose input data points already represent useful information and require no formal front-end processing.

Many perceptual frameworks [4, 5, 6] implement the reprocessing concept only as reconfiguration guided by differential diagnosis. Often, they continuously gather data from every available SPA whether required for interpretation improvement or not. Only when ambiguous data is observed are certain SPAs' outputs actually examined to distinguish between competing interpretations. This approach's uncertainty representations attribute deviations between signal behavior and event models solely to chance source variations, never to a signal's interaction with unsuitable SPAs.

In the GOLDIE system [11] interpretation goals guide the choice of image segmentation algorithms, their parameter settings, and their application regions within an image array. The system generates symbolic explanations for an algorithm's (un)suitability to a particular region. In these features the framework approaches the capabilities of IPUS, but notably it does not incorporate diagnosis. If an algorithm's segmentation is unexpectedly poor, the system cannot diagnose the result and use this information to reformulate algorithm search, but simply re-segments with the original search's next rated algorithm.

Conclusion

IPUS provides structured, bidirectional interaction between the search for SPAs appropriate to the environment and the search for interpretations to explain the SPAs' output. The availability of a formal signal processing theory is an important criterion for determining the architecture's applicability to a domain. IPUS allows system developers to organize signal processing knowledge into formal concepts of discrepancy tests, SPA processing models, distortion operators, and reprocessing-SPA application strategies. A major architectural contribution is to unify SPA reconfiguration performed for symbolic-based interpretation processes with that performed for numeric-based processes as a single reprocessing concept.

With respect to scaling, one might argue that the time required by multiple reprocessings under IPUS would be unacceptably high in noisy environments. This view ignores IPUS's advantage over other paradigms in that it *selectively* samples several front-end processings' outputs, avoiding the traditional approach of continuously sampling several front-end processings' results. IPUS also encourages the development of fast, highly specialized, theoretically sound SPAs for reprocessing in appropriate contexts [13]. In this respect the IPUS paradigm decreases the expected processing time for scenarios requiring several processing views for plausible interpretations.

Our acoustic testbed experiments indicate that the basic functionality and interrelationships of the architecture's components are realizable. An indication of the architecture's generality can be seen in its use not only in the acoustic interpretation testbed discussed in this paper but also in a radar interpretation system being developed at Boston University. Our current work in the architecture is concerned with predicting bounds on the amount of reprocessing an environment can induce in IPUS-based systems.

Acknowledgments

We would like to acknowledge Norman Carver for his role in developing IPUS's control framework and evidential reasoning capabilities. Malini Bhandaru and Zarko Cvetanović were important contributors to the testbed's early implementation stages, and Erkan Dorken was an important contributor to the design of testbed SPAs.

References

[1] Allen, J. F.; Hayes, P. J. "A Common-Sense Theory of Time," *Proc. 1985 Int'l Joint Conf. on AI.*

[2] Bitar, N.; Dorken, E.; Paneras, D.; and Nawab, H. "Integration of STFT and Wigner Analysis in a Knowledge-Based Sound Understanding System," *Proc. 1992 IEEE Int'l Conf. on Acoustics, Speech and Signal Processing.*

[3] Carver, N. "A New Framework for Sensor Interpretation: Planning to Resolve Sources of Uncertainty," *Proc. 1991 AAAI.*

[4] Dawant, B.; Jansen, B. "Coupling Numerical and Symbolic Methods for Signal Interpretation," *IEEE Trans. Systems, Man and Cybernetics.* Jan/Feb 1991.

[5] De Mori, R.; Lam, L.; Gilloux, M. "Learning and Plan Refinement in a Knowledge-Based System for Automatic Speech Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Feb 1987.

[6] Dove, W. *Knowledge-Based Pitch Detection*, PhD Thesis, Computer Science Dept., MIT, 1986.

[7] Dorken, E.; Nawab, H.; Lesser, V. "Extended Model Variety Analysis for Integrated Processing and Understanding of Signals," *Proc. 1992 IEEE Int'l Conf. on Acoustics, Speech and Signal Processing.*

[8] Erman, L.; Hayes-Roth, R.; Lesser, V.; Reddy, D. "The Hearsay II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty," *Computing Surveys*, v. 12, June 1980.

[9] Hayes-Roth, B.; Washington, R.; Hewett, R.; Hewett, M.; Seiver, A. "Intelligent Monitoring and Control," *Proc. 1989 Int'l Joint Conf. on AI.*

[10] Hudlická, E.; Lesser, V. "Meta-Level Control Through Fault Detection and Diagnosis," *Proc. 1984 AAAI.*

[11] Kohl, C.; Hanson, A.; Reisman, E. "A Goal-Directed Intermediate Level Executive for Image Interpretation," *Proc. 1987 Int'l Joint Conf. on AI.*

[12] Lesser, V.; Nawab, H.; et al. "Integrated Signal Processing and Signal Understanding," TR 91-34, Computer Science Dept., University of Massachusetts, Amherst, MA, 1991.

[13] Nawab, H.; Dorken, E. "Efficient STFT Approximation using a Quantization and Difference Method," *Proc. 1993 IEEE Int'l Conf. on Acoustics, Speech and Signal Processing.*

[14] Nawab, H.; Lesser, V. "Integrated Processing and Understanding of Signals," ch 6, *Knowledge-Based Signal Processing*, A. Oppenheim and H. Nawab, eds, Prentice Hall, New Jersey, 1991.

[15] Nawab, H.; Quatieri, T. "Short-Time Fourier Transform," *Advanced Topics in Signal Processing*, Prentice Hall, New Jersey, 1988.

[16] Nawab, H.; Lesser, V.; Milios, E. "Diagnosis Using the Underlying Theory of a Signal Processing System," *IEEE Trans. Systems, Man and Cybernetics*, Special Issue on Diagnostic Reasoning, May/June 1987.

[17] Nii, P.; Feigenbaum, E.; Anton, J.; Rockmore, A.; "Signal-to-Symbol Transformation: HASP/SIAP Case Study," *AI Magazine*, vol 3, Spring 1982.

[18] Peng, Y.; Reggia, J. "Plausibility of Diagnositic Hypotheses: The Nature of Simplicity," *Proc. 1986 AAAI.*

[19] Swain, M.; Stricker, M. eds. *Promising Directions in Active Vision,* NSF Active Vision Workshop, TR CS 91-27, Computer Science Dept, University of Chicago, 1991.