AI Magazine Volume 7 Number 3 (1986) (© AAAI)

Brant A. Cheikes

Research in Artificial Intelligence at the University of Pennsylvania

Overview

The Department of Computer and Information Science (CIS), in the University of Pennsylvania's School of Engineering and Applied Science, conducts a wide range of research projects in artificial intelligence and related disciplines. Participating in this research effort are not only the CIS faculty and students but also members of the chemical, civil, electrical, mechanical, and systems engineering departments; the linguistics, philosophy, and psychology departments; the School of Medicine; and the Wharton School. Primary research interests include natural language processing, knowledge representation, expert systems, automated reasoning and logic programming, analysis and synthesis of motion, computer graphics, graphics animation, program specification and transformation, parallel processing for AI systems, computer vision and robotics, integration of visual and tactile perception, real-time distributed operating systems for vision and robotics, and medical imaging.

Research Computing Facilities. Facilities available for research computing include VAXes, Symbolics LISP machines, MicroVAXes, Sun and Hewlett-Packard workstations, high-speed graphics displays, and special-purpose vision and robotics equipment. In addition, some parallel computers are expected to be installed later in the year. The CIS Research Computing Facility, consisting of six VAX computers running VMS and arranged in a VAX-cluster, provides general computing services to all CIS and CIS-affiliated researchers. Besides these general facilities, individual laboratories maintain their own equipment (to be described in more detail later), with all CIS computers connected by an Ethernet. The CIS research facilities are managed by Ira Winston, with assistance from Suzanne Morin.

The CIS department maintains connections to CSNET (node UPENN) and ARPANET for exchanging information with other universities and research institutes.

Research Laboratories. The bulk of the research conducted in the CIS department takes place in four laboratories: the Language, Information, and Computation (LINC) Laboratory; the Computer Graphics Laboratory; the General Robotics and Active Sensory Processing (GRASP) Laboratory; and the GRASP Image Processing (GRIP) Laboratory. This report first describes the research being conducted in the various labs, then the research carried on outside the lab environment.

The LINC Laboratory

The LINC laboratory is the center for research in natural language processing, expert systems, knowledge representation, automatic theorem proving, and logic programming. The faculty members supervising projects in this laboratory are Aravind Joshi and Bonnie Webber (natural language processing), Tim Finin (knowledge representation and expert systems), Lokendra Shastri (knowledge representation), and Dale Miller (automatic theorem proving and logic programming). Technical support for the lab is provided by Richard Billington.

Abstract

This report describes recent and continuing research in artificial intelligence and related fields being conducted at the University of Pennsylvania. Although AI research takes place primarily in the Department of Computer and Information Science (in the School of Engineering and Applied Science), many aspects of this research are performed in collaboration with other engineering departments as well as other schools at the University, such as the College of Arts and Sciences, the School of Medicine, and the Wharton School.

Brant A Cheiks is affiliated with the Department of Computer and Information Science, The Moore School of Electrical Engineering, University of Pennsylvania, Philadelphia PA 19104

I would like to thank all the faculty members who put up with my initial and follow-up interviews, as well as those who wrote or edited portions of this report; it would have been impossible to write this report without their time and interest I would like to especially thank Ellen Hays for the many ways she helped make this paper more readable Finally, I would like to express my appreciation to Aravind Joshi and Bonnie Webber for their thoughtful critiques of various drafts of the report

Research facilities available to the LINC lab include a VAX-11/785 (running Ultrix), ten Symbolics LISP machines (a combination of 3600s, 3640s, and 3645s), twenty Hewlett-Packard Bobcat 9000/300 AI development workstations[1] (of which eight are in the LINC lab proper and twelve are distributed throughout various other labs for expert system development work), a Sun workstation, and an Apple Macintosh.

Research in Natural Language Processing

LINC lab researchers in natural language processing have been focusing their attention on the design of cooperative interactive systems, such as database query systems and expert advisory systems (Finin, Webber, & Joshi, 1986), and the computational properties of grammatical formalisms for expressing the syntactic, semantic, and (ultimately) pragmatic meaning of natural language utterances.

Interactive Systems

With computer systems becoming increasingly popular both for their ability to maintain large and complex databases and for their usefulness in expert advisory capacities, the need for such systems to be capable of sophisticated conversational interaction has become clear. The desire to allow natural language communication between system and user is motivated by the recognition that such a capability can substantially ease the information-seeking process (that is, users can ask their questions in a natural manner, rather than in a structured, formal query language). Furthermore, a system's ability to understand and respond in a natural language greatly reduces or eliminates the need for training users in system operation.

Unfortunately, as computer systems are able to perform increasingly complex activities, the kinds of interaction such systems must support increase accordingly. Research in the LINC lab has been geared toward identifying properties of interaction deemed necessary in cooperative systems and developing computational methods by which they can be implemented. Several recent projects are described in the following sections.

Providing Definitions of Concepts. When database systems are large and complex, it is generally undesirable to assume that users know the system's structure and content or that they understand the terms and concepts stored in the database. Recognizing this, Kathleen McKeown devoted her Ph.D. thesis (McKeown, 1985) to developing a domain-independent technique by which a system can generate natural language explanations of the terms it uses. Her system, called TEXT, works by "piggybacking" its definitional machinery on an augmented version of a database's domain model. When a user asks for a definition of a term or for an explanation of the difference between two terms, TEXT does the following:

- Collects all the relevant information it might include in its response (not all of this information will necessarily appear in the actual response, but no information other than this will appear)
- Chooses an appropriate, fairly flexible format (called a *schema*) for presenting the information to the user
- Fills in the schema with a subset of the relevant information, guided by the options allowed in the schema and rules that constrain the focus of the text to change in a natural way
- Generates natural language text from the instantiated schema

In the past year, TEXT has been ported to the Symbolics LISP machine environment (it was originally written in Franz LISP on a VAX) and connected with McDonald's text generator MUMBLE (McDonald, 1983), which was written at the University of Massachusetts and adapted locally to work with TEXT (Rubinoff, 1986). Input to TEXT now comes from the RUS parser, imported from Bolt, Beranek, and Newman, Inc. (BBN). Future work with TEXT involves grounding it in a richer knowledge representation system which should enable it to produce definitions that are better tailored to the specific situation.

Although TEXT was designed to provide definitions of concepts known to a database, Robert Rubinoff developed the CLEAR system (Rubinoff, 1985) to do the same for concepts contained in an expert system. CLEAR bases its response on information extracted from the expert system's rule base and requires little or no augmentation of the system to generate explanations. In deciding whether to use a rule in its description of a concept, CLEAR ranks the rule according to where and how the concept is used in that rule. Based on these rankings, CLEAR selects the rules that are most useful for explaining the concept and then uses templates for generating the actual response.

Correcting Misconceptions. Another possible impediment to successful interaction between user and system involves discrepancies between the system's beliefs about the world and those of the user. Many difficulties in interaction arise because the participants hold different views about the world and either do not realize it or fail to do anything about it. As a result, they may leave the conversation with different views regarding what was communicated.

In a recently completed Ph.D. thesis, Kathleen McCoy developed a method for correcting object-related misconceptions (McCoy, 1985). Such misconceptions may be revealed when a user's utterance describes an object in terms of a class it doesn't belong to, incorrectly attributes some property to it, or ascribes an incorrect value to one of its properties. McCoy's system, called ROMPER (Recognizing Object-related Misconceptions using PERspec-

[1]Twenty more Bobcats are due over the next two years.

tive), uses a technique called *perspective* to identify possible object-related misconceptions, then selects from a variety of possible correction strategies based on what the system believes led the user to the misconception.

Preventing Misconstrual. It has been shown by several researchers that the direct answer to a question can often mislead the hearer into believing something the speaker knows to be false (Joshi, Webber & Weischedel, 1984a; Joshi, Webber & Weischedel, 1984b; Kaplan, 1982; Mays, 1980.) Joshi (Joshi, 1982) has suggested that if a speaker is about to say something the speaker believes could lead the hearer to draw a false inference, the speaker is obliged to provide additional information to prevent such an inference.

According to Grice (Grice, 1975), one method of communicating information to a listener is through *implicature.* For her doctoral thesis, Julia Hirschberg has developed a theory of *scalar implicatures*, formulated a representation for them, and built a system that can identify scalar implicatures in an utterance (Hirschberg, 1985). Such a system could be used to identify aspects of an utterance that might cause a hearer to draw a false inference, at which time the response could be modified or extended so as to block that inference.

Modeling the Speaker's and the Hearer's Beliefs. Martha Pollack (currently at SRI) has completed Ph.D. research in which she developed a model of questions and responses that does not rest on what she calls the *appropriate query assumption* (Pollack, 1985; Pollack, 1986). She abandons the notion that the speaker and the hearer share the same beliefs about the world and accordingly suggests a model of plan inference that is significantly different from the models discussed by Allen (Allen, 1982), Cohen (Cohen, 1985), and others. Where their inference procedures are essentially heuristically-guided searches through the space of valid plans, Pollack's procedure involves reasoning about likely differences between the speaker's own beliefs and those of the hearer, then searching through the space of plans that could be produced by such differences. To facilitate this reasoning, Pollack has had to re-analyze the notion of plans, giving a careful account of the mental attitudes entailed by having a plan.

The LandScan Project. Ellen Hays is developing a natural language front end to the LandScan (LANguage-Driven SCene ANalyzer) project, which involves members of both the LINC and GRASP labs. LandScan takes a unique approach to the problems of visual object recognition. Recognition and analysis of objects in a scene are performed dynamically in response to user queries; this mostly eliminates undirected computation by the vision system. From the point of view of natural language research, the interesting problem is how to map what might look like a simple database query to directions that will guide the vision system in its tasks. Currently, Hays is analyzing the domain of urban scenes (the project uses a scale model of part of the University of Pennsylvania campus for testing purposes) and trying to implement a grammar (for use by the RUS parsing system) to interpret the kinds of questions the system should be able to answer.

Planning and Temporal Reasoning. In a past Ph.D. thesis, Eric Mays developed a modal temporal logic for reasoning about necessity and possibility (Mays, 1984). Mays saw the need for a technique for computing the usefulness of a particular kind of response behavior he called a *monitor offer.* The work was implemented in a tableau-based theorem prover, which is available on the Symbolics LISP machines for experimentation.

Extending this work, Brant Cheikes has studied several issues concerning planning and temporal reasoning capabilities needed by question-answering systems to generate relevant responses (Cheikes, 1985). Currently, Cheikes and Yves Schabes are developing a representation for plans, events, and actions that incorporates time explicitly and makes it possible to reason about uncertain events. One goal of this effort is to build a system that can evaluate the viability of plans and correct them if necessary.

Interactive System Architecture. In the past, researchers both at the university and elsewhere have studied a wide variety of behaviors deemed necessary in interactive systems, and most have implemented their work in computer programs. Few, however, have ever attempted to design systems capable of demonstrating combinations of behaviors. As a result, little is known about the problem of actually building useful interactive systems, though most of the subproblems have been studied at least once.

On the assumption that attempting to formulate the architecture of an interactive system will indicate new areas for study (as well as being a research problem in itself), Cheikes is working on the structured design of a modular, flexible architecture for a question-answering system. It is hoped that this architecture will ultimately serve as the foundation of a coherent system into which the implemented products of future research can be integrated and will provide a framework within which future LINC lab natural language research can be conducted.

Studies of Grammatical Formalisms

Aravind Joshi, along with several graduate students (Sunil Shende, David Weir, Vijay-Shanker, and John Dowding), is investigating the computational properties of a particular grammatical formalism called *tree-adjoining grammars* (TAGs). TAGs were invented by Joshi and his colleagues in the late 1970s (Joshi, Levy, & Takahashi, 1975), and have recently become the object of much interest in the linguistics community. One interesting property of TAGs is that they appear to be able to capture cross-serial dependencies, a feature not generally found in English but prevalent in languages such as Dutch and Swiss German. Vijay-Shanker and Weir have shown that TAGs bear a close relationship to another grammatical formalism called head grammars (due to Carl Pollard at Stanford University (Pollard, 1984)), and have done some work that seems to indicate that TAGs are a natural next step up in complexity from context-free languages (Vijay-Shanker & Joshi, 1985; Weir, Vijay-Shanker & Joshi, 1986a; Weir, Vijay-Shanker & Joshi, 1986b).

Shende is exploring the application of parallel processing to the problem of parsing natural language using TAGs. Dowding is working on TAG parsing in a logic programming framework, and Billington and undergraduate Andrew Chalnick are building a graphics editor for TAGs on the Symbolics LISP machine. In the linguistics department, Tony Kroch is examining the linguistic ramifications of TAGs (Kroch & Joshi, 1985).

Discourse Analysis

When people talk or write, they refer to things, objects, events, actions, facts, or states that have been mentioned before. Such context-dependent reference is called *anaphora*. In general, linguists and AI researchers have looked at the problem of anaphora interpretation as a search for the correct antecedent for an anaphora, that is, the previous words or phrases to which the anaphora is linked. Lately, people studying anaphora have suggested that in order for anaphors to be resolved correctly, they must be linked to entities evoked by the previous discourse rather than to their grammatical antecedents (Webber, 1982; Kamp, 1984).

For her doctoral thesis, Ethel Schuster is working on a technique for dealing with anaphoric language when the reference is to events and actions. It involves four issues:

- What aspects of the discourse give evidence of the events and actions the speaker is talking about
- How actions and events are represented in the listener's discourse model
- How to identify the sets of events and actions as possible choices
- How to obtain the speaker's intended referent to an action or event from a set of possible choices

Anaphoric forms that are used to refer to actions and events include *sentential-it* and *sentential-that* pronominalizations as well as *do it*, *do that*, and *do this* forms. Because the anaphoric forms have many possible interpretations, linguistic information alone is insufficient for understanding them; models of the discourse are needed as well.

In related work, Barbara Grosz (SRI), Aravind Joshi, and Scott Weinstein (philosophy department) are investigating the role of pronouns and definite descriptions for local coherence of discourse (Grosz, Joshi, & Weinstein, 1986).

Automating Database Design

For his Ph.D. thesis, Sitaram Lanka is studying methods of automating the design of database schemas. The state of the art in database design is such that to create a database for a particular application, specifications must be provided using certain concepts and notations specific to the database management system. The user or designer creating the database thus has to deal with concepts that are foreign to the application domain. Even for an expert, this is a tedious and error-prone task. Ideally, the designer should only be concerned with providing a picture of how the database is expected to be used. Lanka's approach involves inferring the database schema from a series of natural language queries to be answered by the database designer. His work uses the *functional data model* to express the database schemas and draws heavily on ideas from the Knuthian semantic system (attribute grammars), a technique for specifying semantics of languages defined by a context-free grammar. A prototype system has been implemented in Common LISP on the Symbolics LISP machine.

Research in Expert Systems

Tim Finin is directing two research projects in expert system design. The first project, called the Toxic Waste Advisor, is being carried out with support from the civil engineering (CE) department and involves CIS graduate students Mark Beutnagel and Joung-Woo Kim, CE faculty members Iraj Zandi and John Keenan, and engineers from the Weston Corporation. The goal of the project is to design a system to advise engineers on the cleanup of toxic waste sites. A variety of techniques exist, each of them more or less useful depending on the characteristics (such as location and geographic topology) of a particular site. The expert knowledge in this area is being provided by Joseph Kolmer, a vice president of the Weston Corporation. At present, a prototype system is being developed using the Texas Instruments Personal Consultant expert system development tool.

The second project concerns the development of an interactive computer-aided design and computer-aided manufacturing (CAD-CAM) system for the engineering design of injection-molded plastic parts and involves mechanical engineering (ME) faculty members Mike McCarthy and Martin Lesser and two ME graduate students. There are many problems and interactions between the design phase of a molded plastic part and its eventual fabrication, some of which affect the part's ultimate cost or its ease of manufacture. It is envisaged that the expert system currently being designed will provide feedback to the part designer in order to anticipate and avoid problems that could arise at later manufacturing stages. This research is being carried out with funding from the Scott Paper Corporation.

In addition, Bonnie Webber is collaborating with Dr. John Clarke of the Medical College of Pennsylvania in developing a computer-aided decision support system to support medical corps personnel in the management of specific accidental injuries. The system will be used to give management advice to medical corps personnel in isolated environments. Advice will support decisions on whether to evacuate the patient, the urgency of the evacuation, emergency measures necessary while awaiting evacuation, and the definitive management of injuries not requiring evacuation. The point of the project is to develop methods whereby an expert system developed for one group of users with a particular set of resources and in a particular environment (for example, emergency room physicians) can be transformed systematically for use by another group with different resources and in a different environment in dealing with the same problem.

- **Real-Time Expert Systems.** Tim Finin and graduate student David Klein are developing methods to accommodate very large, frequently modified knowledge bases in real-time intelligent control systems. Although the effective handling of large knowledge bases is a general problem for AI systems, opportunities exist for self-tuning and knowledge verification that are generally unavailable or impractical in consultative expert system environments because intelligent control systems communicate directly with their environments.

Work in this area focuses on the following:

- Integrating mechanistic models of the target environment with domain-independent inference mechanisms and "standard" knowledge repositories for real-time intelligent control
- "Compiling" inferences (that would normally be computed in real time) in advance of actual problem occurrences
- Automatically verifying (internally represented) mechanistic models against observations of the real environment
- Interleaving monitoring, reasoning, execution, compilation, and verification to optimize real-time performance

Domains of interest in this research include computer installation operations, chemical process plant management, and communication network management.

Lyle Ungar of the chemical engineering department is particularly interested in real-time expert systems for adaptive control of chemical processes, such as oil refining. For example, the choice of a proper strategy for controlling the temperature of a chemical reaction can make the difference between a greater or lesser yield of the desired chemical product. Ungar is looking at ways of using causal reasoning and teleologic analysis to diagnose equipment failures and to dynamically alter process control strategies. A prototype system for choosing among strategies for controlling the temperature of a reactor has been built and used to control a real-time numerical simulation.

Research in Knowledge Representation

Research in knowledge representation follows both the classic hierarchical representation system approach, such as that used by the KL-ONE knowledge representation language (Brachman, 1978; Brachman, 1985), and the massively parallel interconnection network encoding approach.

Taxonomic Representation. Graduate students Robert Kass and Ron Katriel are collaborating with Tim Finin in exploring various ways of extending the KL-ONE knowledge representation language. A problem perceived in KL-ONE is its handling of so-called "primitive" or "natural kind" concepts—concepts that cannot be completely defined in terms of their roles and inherited attributes. Kass and Katriel are developing strategies for representing partial descriptions of primitive concepts using frame-based representation languages (Kass, Katriel, & Finin 1986). Part of the motivation for this research comes from the desire to build *interactive classifiers*, which are programs that can automatically position a newly defined concept in a concept hierarchy, asking the user for help as needed. This work is supported by a grant from Digital Equipment Corporation.

Connectionist Networks. Connectionist network theory has recently been the focus of considerable attention in knowledge representation and machine-learning circles. *Connectionism* is a massively parallel approach to AI and is motivated in part by theories of how animal neural networks operate. Under the connectionist network paradigm, it is possible to encode computations and solutions to problems in terms of massively parallel networks of highly interconnected but simple processing elements.

Lokendra Shastri has been studying the application of connectionism to knowledge representation, specifically to the representation of conceptual knowledge. He has shown that certain troublesome issues in semantic network theory, such as multiple inheritance and exceptions, have relatively natural solutions within an evidential (probabilistic) framework and that these solutions can be computed efficiently by a connectionist network (Shastri, 1985a; Shastri, 1985b; Shastri & Feldman, 1985).

In general, there are two approaches to research in connectionist network theory: a machine-learning approach and a "tailored network" approach. In the machine-learning approach, a rather general network topology is chosen along with a learning rule for the network. Next, the network is shown a series of examples of inputs and desired outputs. Using the learning rule, the network adjusts itself over time so that the network will eventually produce the desired output when given only an input. The advantage to the machine-learning approach is that network topology is relatively unimportant; however, the training process often requires hundreds of thousands of examples before the desired level of learning is achieved.

The tailored network approach attempts to form fit the network to the problem. Given a problem, some particular network topology, combined with particular operations to be performed at the individual nodes, will provide a solution to the problem. Unfortunately, there is no well-developed theory of programming connectionist networks, and the problem of choosing a topology and an appropriate set of processor functions is difficult.

Shastri and graduate student Ron Katriel are examining yet a third approach. This hybrid approach begins with a network that has some innate structure and applies learning rules to it. Here it is assumed that the innate structure reflects some general characteristics of the problem domain and corresponds to what might be called the result of evolutionary learning. In this sense, the changes in the network resulting from the application of learning rules during training correspond to developmental learning. It is hoped that such a strategy might lead to better insights into the problem of learning in general and learning in connectionist networks in particular. This strategy is being investigated using the game of checkers as a test bed.

Research in Logic Programming and Theorem Proving

Two important topics of research in the LINC lab are logic programming and automatic reasoning. Logic programming work concentrates on the design of a more powerful logic-based language and automatic reasoning work focuses on the implementation of an interactive instructional theorem-proving tool.

Higher-Order Logic Programming. Dale Miller and graduate student Gopalan Nadathur are completing the design and implementation of a higher-order version of the Prolog logic programming language (Nadathur & Miller, 1986a). In this language, called λProlog, function and predicate variables are permitted and can be instantiated with typed λ-terms. Higher-order unification is used to compute such λ-terms. As a result, the language supports many LISP-like, higher-order operations, such as function mapping and application, in a well-founded theoretical fashion.

The λProlog language has been shown to have applications to natural language processing. Many aspects of theoretical linguistics are commonly represented and reasoned about within a higher-order logic framework. However, actual programming or representation systems used in computational linguistics and knowledge representation are either not based on logic or are based on a first-order logic. This is unfortunate because it is often difficult to bridge the gap between an actual computational system and its formal justifications.

Miller and Nadathur have shown how higher-order logic can explicitly specify various linguistic programs in such a way that their theoretical justifications are direct and natural (Nadathur & Miller, 1986b). They have shown how knowledge representation schemes (similar to frames or KL-ONE concepts) can be directly encoded as λ-terms and how the computational process of determining subsumption can be written simply as a program in λProlog. Because their formalism is based solely on logic (albeit a higher-order logic), it is possible to integrate smoothly the semantic (knowledge representation) and syntactic (definite clause grammar parsing) components of a natural language understanding system.

Interactive Theorem Proving. Miller and graduate students Amy Felty and Greg Hager are exploring the issues involved in building computer systems that can integrate both interactive and automatic theorem proving in a variety of logical languages. Within such a system, several different kinds of proof systems must be supported because some, such as resolution, are convenient for automatic theorem proving, and others, such as natural deduction, are useful for interactive theorem proving. At the heart of this integration is an ability to automatically transform proofs from one of these systems into another. Hence, it is possible for a resolution refutation of a theorem to be transformed to a more readable natural deduction proof. Felty has designed and implemented many of the necessary proof-transformation algorithms for first-order logic (Felty, 1986). Hager's M.S.E. thesis (Hager, 1985) extended several of these algorithms and proof systems to various modal logics. Similar research has also been done to accommodate higher-order logic.

The goals of this work are two fold. First, through the use of sophisticated manipulations of proofs, it should be possible to build human-oriented natural language explanation systems of computer-generated proofs. Such a system must be able to plan the high-level rhetorical structure of an explanation and then manipulate the given abstract proof to fit into that plan. The current work on constructing natural deduction proofs is a first step in this direction. The second goal is to provide a tool with which researchers and students can experiment with various logical languages and proof systems. A prototype version of such a tool has been built for first-order logic. Its interactive theorem prover, an extension of the LCF (logic of computable functions) class of theorem provers, is extensible and programmable, thus providing for a wide range of interaction. It also contains a resolution-style automatic theorem prover that is integrated with the interactive theorem prover using the proof transformations mentioned earlier. The integrated approach allows both trivial and nontrivial subproofs of a given natural deduction proof to be solved by resolution and then automatically translated back into natural deduction for the user.

The Computer Graphics Laboratory

The Computer Graphics Laboratory, under the direction of Norman Badler, is studying the production and graphic display of controllable human figures engaged in performing tasks. Research is principally carried on under the aegis of the TEMPUS project, which is developing an interactive system for graphically simulating the activities of several human agents in a three-dimensional (3-D) environment. Related projects involve facial animation, the design of knowledge bases that can represent the functionality of 3-D objects, the development of new techniques for graphical manipulation of 3-D objects, and the development of an interactive graphics interface designer.

Research in the Graphics lab is carried on by nearly 30 students (most of them graduate students), one faculty member (Badler), and two full-time staff members (Jon Korein and Craig Meyer). The research is conducted using a variety of machines, including a VAX-11/785, two frame buffer systems (one Grinnell frame buffer and one Lexidata SV-2 Solidview), an IRIS workstation for real-time color graphics display, a six-dimensional (6-D) digitizer, and an assortment of other filming and display devices.

Modeling Natural Human Movement

Through their various research projects, workers in the Graphics Lab are attempting to develop an understanding of the characteristics of human motion. Motion performance, observation, description, and notation all influence the form of a motion representation. Because a motion representation can be verified by a computer graphics performance, the effective control of natural-looking human figure movement is a significant and challenging goal.

In a recent report, Badler discusses the characteristics of a computationally realizable human movement representation, which includes distinctions between hierarchic levels, kinematics, and dynamics (Badler, 1986). The qualitative factors of Effort-Shape notation are used to suggest extensions to existing movement representations in directions consistent with known characteristics of human movement and conventional animation. With the proposed representation, useful and expressive motion qualities can be approximated using a combination of kinematics and dynamics computations, with kinetic control modulated by accelerations and decelerations derived from existing interpolation methods. Interactions between motions by phrasing, temporal property, or relationship can be described and executed within an appropriately detailed model.

The TEMPUS Project

The TEMPUS project is a natural outgrowth of the lab's interest in human body modeling. Its immediate goal is to allow a user to model a 3-D environment, a set of human agents, and tasks for the agents to complete. Principally funded by the National Aeronautics and Space Administration (NASA), TEMPUS was conceived as a task simulation facility by the Crew Station Design Section of the Johnson Space Center for the evaluation of complex workstations (Badler *et al.* 1986). Most of the previous research for the TEMPUS project has dealt with effective and efficient algorithms for the graphical rendering of the human figure (using spheres or polyhedra) (Badler & Bajcsy, 1978; Badler, O'Rourke, & Toltzis, 1979) and computational aspects of "reach" (Korein, 1985). This work has formed the foundation for the higher-level studies described in the following subsections.

HIRES: Hierarchical Reasoning System. Part of the usefulness of the TEMPUS system is its ability to simulate the actions of multiple agents in a 3-D environment. Such a simulation should support not only the study of processes over time but also simulation over multiple levels of abstraction.

A hierarchical reasoning system called HIRES, developed by Paul Fishwick as part of his Ph.D. research (Fishwick, 1986), permits modeling of a process or activity at multiple levels of detail or with differing representations. Though implemented as a rule-based system, HIRES supports preprocessors for continuous system simulation models, discrete simulation models, Petri nets, timed Petri nets, and scripts. For example, an elevator simulation might be represented at the lowest level as a dynamics problem, that is, by modeling within a continuous system simulation paradigm; the next level might be a (discrete) queuing system model; and the upper levels can be symbolic, state-based, qualitative reasoning models. A common knowledge base and mechanisms for moving up and down through the hierarchy of models are provided. The resulting simulation creates explicit motion parameters (for example, from the dynamics level) or parametric key frames (from the discrete modeling levels). HIRES has been used to animate five human figures seated at a table, eating according to the well-known "dining philosophers" problem in concurrent processing. The sequence and synchronization tasks are handled in HIRES; the motions are generated by the execution of the simulation at the desired process abstraction level. The eating process itself is modeled as a set of parametric key frames derived from the human body positioning component of TEMPUS. The simulation selects the appropriate key frames according to the "rules" of the process description.

Task Description and Parsing. Tasks that are to be carried out on a space flight are described in detail (by NASA) in the form of task checklists. Part of the goal of the TEMPUS system is to take these checklists as input, understand them, and then graphically simulate the activities they denote in a given 3-D environment. As a first step, Jeff Gangel has built a natural language interface and a knowledge base that encodes an understanding of some simple English verbs of motion (Gangel, 1985). The output of this interface is passed to HIRES for simulation. This system is being extended to handle more complex action verbs.

Facial Expression Animation. In a recently completed Ph.D. thesis (Platt, 1985), Steve Platt studied the structure and properties of the human face. The face is of particular interest both as a physical object and as a means of human communication. Clearly, the ability to animate the face is an important component of a general human body animation system.

The system Platt designed, called OASIS (Object Action SImulation System), is a facial animation system based on an object-oriented approach to motion control. Its salient feature is the separation of actions and objects from the application algorithm that maps specific actions to specific objects. Though the underlying motions can be defined parametrically, the control flow is organized according to the allowable actions applied to objects and the propagation of those actions or their consequences to other objects within the same moment. For facial animation, this system uses face regions with knowledge of their connections to other face regions—the facial actions know how to propagate across the region connection types. The facial model is general and is structured so as to be capable of molding itself to a particular person's features with a minimum of effort.

Manipulation of 3-D Objects

David Baraff, an undergraduate, is working with Badler in developing new techniques for manipulating the display of 3-D objects by means of a 6-D digitizer and a real-time graphics display. Current input devices, such as light pens or electronic tablets, are unwieldy for describing and manipulating 3-D objects. The researchers aim to develop software and visualization techniques that make it relatively easy to enter descriptions of objects in space.

GUIDE: Graphical User Interface Development Environment

For her Ph.D. thesis, Tamar Granor has developed an interactive system for designing and generating graphical user interfaces (Granor, 1986). Her system, called GUIDE, is an interaction methodology that permits the design and implementation of a User Interface Management System (UIMS) generator. Using GUIDE, an interface designer can interactively specify the user interface for an application. The GUIDE methodology attempts to make interfaces implementable, modifiable, flexible, and consistent and to allow for user variability and application diversity within a user community.

The features of GUIDE are these:

- The designer need not write any interface code
- The designer provides *action routines* that implement the actions or operations of the application system (action routines can have parameters)
- The designer is able to specify multiple control paths based on the state of the system and a profile of the user
- Inclusion of help and prompt messages is as easy as possible
- GUIDE can generate its own interface

GUIDE goes beyond previous efforts in UIMS design in the full parameter specification provided for application actions, in the ability to reference application global items in the interface, and in the pervasiveness of conditions throughout the system. A parser is built into GUIDE to parse conditions and to provide type checking.

Representing Functionality of 3-D Objects

For manipulating 3-D objects in a simulation, a considerable amount of information is needed about the capabilities of the environment as well as of objects in the environment. To date, knowledge bases are generally limited to the representation of geometry and simple nongraphic attributes. Pearl Pu is extending object representations to include functionality, design specifications, assembly and disassembly processes, repair procedures, input-output relationships, state tables, and other general information traditionally associated with the design process but not explicitly captured.

The GRASP Laboratory

The GRASP laboratory is the home of research in robotics, machine vision, and real-time distributed processing. The theme of the robotics work is the combining of multiple sensory input, especially visual and tactile information, in robot controllers. The faculty members affiliated with this lab are Ruzena Bajcsy (robotics and machine vision), Richard Paul (robot control), Sam Goldwasser (special-purpose architecture for robotics), Insup Lee (operating systems for real-time distributed processing), Hossam El Gindy (computational geometry), and Max Mintz (decision theory modeling). Portions of this section and the following section on the GRIP Laboratory have been extracted from several recent quarterly reports published jointly by the two labs (Goldwasser, 1984; Goldwasser & McCarthy, 1985a; Goldwasser & McCarthy, 1985b; Goldwasser, Hager, & Solina, 1986).

The GRASP lab maintains one VAX-11/785 and two VAX-11/750s (all running Ultrix), three MicroVAX-II systems, and one Symbolics 3600 LISP machine. Other facilities of the GRASP lab include a PUMA 560 with VAL-II; a custom mechanical hand (the Pennsylvania Articulated Mechanical Hand) with tactile sensor arrays and force-sensing wrist; GE P50 and Unimate 2000B industrial robots; two stereo-mounted charge-coupled device (CCD) cameras (380 x 480) with computer-controlled focus, aperture, and zoom on a four-axis servo platform with real-time video digitizer; a multichannel microprocessor-controlled variable-intensity lighting system; a GE Optimation II Vision System; Ikonas (Adage) RDS3001 and RDS3010 color graphics systems with microprogrammable graphics processor; a graphics hard-copy camera for 35mm slide or 16mm cine recording; and numerous other microcomputers and special-purpose peripherals. GRASP lab facilities are managed by Ned Batchelder.

Integration of Multiple Modalities

Robotic systems are being designed and built to perform complex tasks such as object recognition, grasping, part manipulation, inspection, and measurement. In the past, systems have been designed that recognize objects using only a single sensing modality. Unfortunately, single-sensor systems are necessarily limited in their power. The approach taken by the GRASP lab overcomes the inherent limitations of single sensors by integrating multiple sensing modalities (such as passive stereo vision and active tactile sensing) for object recognition. The advantages of multisensor systems for such a task are many. Multiple sensors supply redundant and complementary kinds of data that can be combined to create a more complete understanding of a scene. The need for multisensor systems is becoming more apparent as research continues in distributed systems and parallel approaches to problem solving.

GRASP lab researchers are continuing their longstanding research effort in the design and implementation of a robotic system that combines input from multiple sensors. The domain in which the system operates is that of common kitchen items such as pots, pans, dishes, cups, and utensils. This is a rich domain and contains objects representative of many other domains as well. The objects are planar as well as volumetric, contain holes, and have concave and convex surfaces. They are also decomposable into separate components that have functional meaning: handles are distinct geometric parts that are used for grasping, a cup's central cavity is used to hold liquids, a spout allows one to pour a liquid, a lid covers a cavity. By basing the models of these objects on geometry and topology, the system is extensible beyond this domain. The objects are modeled in a hierarchical manner, which allows the matching process to proceed at different levels with support or inhibition from higher or lower levels of model matching.

The current vision system consists of a pair of stereo-mounted CCD cameras with controlled focus, zoom, aperture, and vergence. They are mounted on a four-degree-of-freedom (X, Y, pan, tilt) camera frame under computer control. The tactile system consists of a one-fingered tactile sensor attached to the wrist of a PUMA 560 robot. The control module is the overall supervisor of the system, and is responsible for guiding and directing the visual and tactile sensing modules. This module also communicates with the model database during the recognition cycle as it tries to interpret a scene. It is able to use both low-level reasoning about sensory data and high-level reasoning about object structure to accomplish its task. Both kinds of reasoning are needed, and the system's ability to alternate between the two kinds of reasoning makes it powerful. The high-level reasoning allows the system to use the object model as a guide for further active sensing, which is then accomplished by the low-level sensing modules.

An active, multisensor intelligent system for object recognition (called ARAMIS for "A Rudimentary Active Multimodal Intelligent System") is being developed by Sharon Stansfield, combining passive stereo vision with active touch (Stansfield, 1985). The major issues addressed by this work are the following:

- The knowledge representation and modeling
- The integration of multimodal data
- The strategies for reasoning and active perception

Stansfield is currently working on identifying the tactile primitives and features that will make up the system and on determining and designing the active exploratory procedures used to isolate them (Stansfield, 1986). Psychologists suggest that the human haptic system computes information related to an object's form, substance, and function. Stansfield accepts this model and further requires that tactile primitives be directly computable or measurable as opposed to derived. Using this definition, she proposes primitives such as compliance, elasticity, point contact, and texture. Such tactile primitives can then be combined into more complex tactile features. It is at this state that some problem-solving abilities are required. She has chosen to implement edges, surface patches, holes, and cavities as her initial features and is testing exploratory procedures (implemented as a set of VAL and C programs) for determining each primitive and feature. The equipment used is a LORD Corporation LTS-200 array-vector sensor mounted on the wrist of a PUMA 560 robot arm.

General Robotics

There are several research projects addressing various issues of general robotic theory.

Mechanical Analysis and Design of Manipulators. Alberto Izaguirre has finished building a LISP system that generates dynamic robot motion equations automatically (Izaquirre, 1986). The program takes as input a file containing all the information about a robot (such as its Denavit-Hartenberg parameters, centers of gravity, masses, and the moments of inertia of its links as well as the characterization of its links as revolute or prismatic). Its output consists of another file containing a C program that calculates the dynamic equations of the robot. The automatic approach is extremely beneficial, because generating the equations manually is time consuming and error prone.

Electromyographic Input. John El-Ward is exploring ways of using electromyographic (EMG) signals from the human arm for teaching the Pennsylvania Anthropomorphic Arm (a robot arm with seven degrees of freedom built in the GRASP lab). The intent is to allow a robot programmer to use motions of the programmer's own arm to generate the sequence of motions desired from the robot. The EMG approach is a potential alternative to the earlier approach of electromechanical instrumentation of the human arm.

Ultrasound Ranging. Peng Woo has implemented closed-loop feedback control of the absolute position of the end effector of an educational robot by means of ultrasound ranging. The robot joints are driven by stepper motors using strings and pulleys. The strings slip on the pulleys sufficiently to cause errors of up to an inch in absolute position of the end effector. A Polaroid ultrasound ranging device mounted on the end effector was modified to permit ranging with an absolute accuracy of $\frac{1}{8}$ inch. The distance from the end effector to a wall was monitored by the ultrasound system, automatically recorded, and compared to the desired distance; the absolute error was used to modify the commands to the robot controller in order to achieve correct absolute positioning.

Current activity on this project focuses on comparison of alternative methods for measuring absolute position in various robot applications and on prediction of the ultimate capability of ultrasound ranging for accurate positioning of robots in 3-D space. Woo is also working with Professor Nelson Dorny of the systems engineering department to design a test-bed system that will ultimately use an industrial robot to catch slow-moving objects using surveillance, tracking, and interception techniques.

System Integration Test Bed. Steve Chao is designing a system integration test bed. The purpose of the test bed is to develop the capability for both instruction and research of integrating all functions associated with manufacturing. The test bed will consist of workstations for computer-aided planning, CAD, computer-aided engineering (CAE), CAM, and computer-aided training (CAT); all functions will be integrated using a computer-integrated manufacturing (CIM) database. Communication will take place through a local area network. The CAM station will include the capability for real-time control of industrial and experimental robots. The test bed should provide a mechanism for connecting and integrating a broad array of research activities from several engineering departments.

Robot Control Theory

Several GRASP lab members are studying problems in robot control theory.

Theory of Grasping. Graduate student Jeff Trinkle recently completed work with Richard Paul developing a theory of frictionless grasping (Trinkle, 1985). Currently, Trinkle is developing a grasp planner in a frictionless, two-dimensional (2-D), polygonal environment (Trinkle, 1986). The planner attempts to position the hand and generate the subsequent finger motions that result in picking up an object and bringing it into contact with the palm.

Effort to date has centered on the accurate quasi-static simulation of the object in contact with the hand. At a particular step in the simulation, the object and the hand touch at a number of points. The instantaneous kinematic constraints due to the contacts are analyzed using linear programming techniques to determine how the fingers can move. After the planner chooses the finger motions, the simulator solves the appropriate system of nonlinear constraint equations to determine the configuration for the next step of the simulation. Once the object contacts the palm, the planner chooses finger motions to minimize the contact forces and joint torques in a gradient-type search procedure.

Universal Servo Controller. Professors Paul and Goldwasser, together with staff member Filip Fuma, are developing a microprocessor system for the generalized control of any robot manipulator using DC motor actuators and optical shaft encoders. This will be accessible over the Ethernet from any GRASP Lab machine, and will provide a highly flexible, easy-to-program environment for the implementation of advanced control algorithms.

Robot Force and Motion Server. Hong Zhang has been working with Richard Paul on a multiprocessor controller for a robot manipulator (Zhang & Paul, 1986). The system will include seven axes using Intel 8086-based single-board computers communicating by direct memory access with another Intel 8086 or 80286 supervisor. The controller provides sufficient computational power for the implementation of complex control algorithms such as manipulator force control; it communicates with other systems a robot depends on, such as its sensors, over an Ethernet so that the controller itself does not need to perform sensory information processing. More importantly, such communication allows programmers to develop and execute robot applications on a UNIX system.

Machine Vision

Because vision can provide a robot with essential data, GRASP lab researchers are studying techniques for actively and effectively extracting information from machine vision systems.

Active Vision. Eric Krotkov has been working on the Pennsylvania Active Camera System (Krotkov, Summers,

& Fuma, 1986). The major mechanical work has been to install gear reductions on the vertical and pitch motors on the camera platform. The major software work has been to debug and improve the lens controller, which controls seven DC servo motors for focusing, zooming, opening and closing the apertures, converging, and diverging.

Krotkov has also been working on software that will allow the camera system to compute the range (distance in an absolute coordinate system) of feature points (Krotkov & Martin, 1986). There are three methods. The first is range from focus, which works by automatically bringing the projection P* of an object point P into sharp focus by maximizing the gradient magnitudes in a neighborhood around P*, then using the laws of geometric optics to compute the distance to P. So far, an accuracy of about five percent of the object distance has been achieved. The second method is range from vergence, which works by triangulation: The distance between the cameras and the angle of convergence are known, allowing the object distance to be uniquely computed by simple trigonometry. The relationship between the position of the vergence motor and the angle of convergence is not precisely known at this time and is being studied. The third method is range from stereo, which converts relative disparities (computed by standard stereo algorithms) into absolute distances by the perspective projection transformation. The missing parameter here is the constant relating pixel size to physical size.

Shape Representation. Franc Solina is working on the problem of category shape representation. Most current vision systems must have a precise model of any particular object they are supposed to recognize. This model might be sufficient in industrial or other restricted environments where the number of objects and their shapes can be controlled. The vision system of a household robot, however, must avoid, approach, or pick up new kinds of objects. Recognition of such new objects can be done on the basis of generic or category shape descriptions. Computer vision research has not yet adequately addressed the problem of recognizing generic objects.

Solina has approached the problem of category shape representation in the context of model-based object recognition. How, for example, can a computer vision system recognize different coffee cups based on a single category model of a coffee cup? Solina's approach is based on functionality and shape deformations. By the principle of functionality, one attempts to understand and explain the influence of an object's function on the shape and composition of its parts. The design of man-made objects must support their intended function; in spite of the wide diversity of forms, certain functional requirements must be satisfied. Deformation of shape, however, is a process that affects both natural and man-made objects. It is a highly intuitive way of describing and thinking about objects. If the most important common property of objects in a category is their function, the shape of categorially related objects must satisfy the same functional constraints. By analyzing these constraints, one can determine a prototypical shape and a set of allowable deformations that account for variations within the category.

Vision and Motion. Knud Henriksen is investigating methods of combining vision and motion. Many people have worked with stereo algorithms (algorithms that can compute depth from a pair of stereo pictures) and motion algorithms (algorithms that can compute the 2-D velocity field in a picture of a moving scene from a series of monocular pictures). In such work, stereo and motion have been considered as two isolated problems.

Experiments are being conducted with feature-based algorithms using line segments as features. The goal is to construct a general matching algorithm that can be used to match line segments both over space (stereo) and over time (motion). The idea is to use a prediction-verification scheme in which the motion information is used to make a hypothesis about the disparity for a given line segment, and the stereo image is then used to verify the hypothesis. A preliminary version of a line finder has been implemented, and the implementation of the matching algorithm is under way. The next step will be to extend the matching algorithm to handle general 3-D motion—both 3-D translation and rotation.

Aerial Imaging. Helen Anderson is working on the LandScan project from the GRASP lab end of the system (see The LandScan Project). She is devising techniques to recognize objects in aerial photographs based on shape and height information. Among the objects to be recognized are buildings, roads, and sidewalks. The recognition system will use data from Erica Liebman's work associating heights with regions from a stereo aerial photograph. The aerial photograph will be of a scale model of the university campus, and the recognition system will be directed by queries from Ellen Hays' natural language front end.

Distributed Processing and Real-Time Systems

Insup Lee is conducting three related projects in the design of systems for real-time distributed processing. The principal focus of research is the Distributed Programming System (DPS), a message-passing-based software system to support the parallel, distributed execution of sequential programs written in various languages, such as C, LISP, and Prolog. For such a system to work, it must hide the differences between different programming languages and different operating systems. For example, when data is passed between computers, data-type conversions must be performed as appropriate.

A DPS prototype was built on top of UNIX by a team of six undergraduates supervised by GRASP lab staff member Batchelder. This work is currently being improved, and graduate student Robert King is building a new distributed real-time operating system (called TIMIX) on MicroVAX-II computers that should efficiently support the kinds of operations needed by DPS. Lee's DPS is being used by other GRASP lab members for implementing distributed robot control systems.

Optical Interconnection Network. Lee, Goldwasser, and student David Smitley are designing an optical interconnection network for interprocess communication called PION (Processors Interconnected with an Optical Network) (David Smitley, 1985). Using lasers, the optical interconnection network can reduce the number of links required for intercommunication between N discrete processors from N^2 to N and still allow any one processor to send a message directly to any other processor. The optical network can be operated in either static or dynamic mode. When operated in the static mode, the network is initially configured to match the topology of the algorithm to be executed and remains in this configuration throughout the computation. Smitley has been concentrating on the development and use of PION in the static mode, specifically trying to devise strategies for determining the optimum process-processor mapping and network topology for a given algorithm. Finding the optimum solution has been shown to be an NP-complete task, so he has developed a heuristic algorithm that finds suboptimal solutions. The heuristic has been analyzed, and its best, worst, and average case behavior has been determined. These results show that the algorithm almost always finds the optimal solution. Even in the situations where a nonoptimal solution is generated, the results are very close to the optimal mapping.

Modeling Real-Time Processing. Lee is working with Ph.D. student Amy Zwarico on formalizing the semantics of real-time processing (Zwarico & Lee, 1985). They are trying to extend Hoare's communicating sequential processes (CSP) model (Hoare, 1985) to include timing constraints on multiple communicating processes. A program is viewed as a composition of processes that interact with an external environment. These interactions can be observed during program execution, obtaining a trace of program execution. The model proposed for real-time programs allows the concise representation of the set of all possible program traces. Each program is modeled by a trace generator that represents the logical sequence of events possible during an execution and the relative time between these events. The trace generator is, however, abstracted from absolute time. The relative timing is captured by the timed eventuality operator found in the work of Bernstein and Harter (Bernstein & Harter, Jr., 1981). The proof system is an extension of the temporal logic systems used in proving concurrent programs correct. The verification of a real-time program consists in showing that the trace generator representing a program is consistent with the specifications of what the program should do.

Computational Geometry

Hossam El Gindy, new to the CIS faculty, is working on problems in computational geometry. His two primary research directions are computing separability of objects in space (with applications to robot control) and designing algorithms for efficient very large-scale integration (VLSI) implementation.

The GRIP Laboratory

The GRIP laboratory is the newest addition to the experimental facilities of the CIS department. The GRIP lab carries on research in algorithms, architecture, software, hardware, and interactive techniques for high-performance image processing and display. Over the past year, the major emphasis has been on the general area of medical-imaging techniques for the real-time display of data derived from computed tomography (CT), positron-emission tomography (PET), and magnetic resonance imaging (MRI) equipment. Sam Goldwasser, along with several graduate and undergraduate students, is at work developing the required specialized hardware and software. The GRIP lab maintains and develops the Voxel-Processor-based physician's workstation and shares most of its research facilities with the GRASP lab, with the addition of a VAX-11/750 and two Adage high-performance graphics systems.

The Voxel Processor

Modern medical-imaging systems such as CT, PET, and MRI generate large quantities of 3-D information, typically in the form of a series of 2-D slices representing cross-sections of the body or head. When these 2-D slices are stacked, a 3-D representation is obtained that contains a wealth of information useful to the physician or medical researcher.

The Voxel[2] Processor, designed in the GRIP lab, represents an architectural framework for the design of systems that provide true real-time interactive manipulation of shaded graphics ($2\frac{1}{2}$-dimensional [$2\frac{1}{2}$-D]) images with full hidden surface removal on a raster-scan cathode ray-tube (CRT) (Goldwasser & Reynolds, 1983a; Goldwasser, 1985a; Goldwasser *et al.* 1985b; Goldwasser *et al.* 1986). Shaded graphics techniques take advantage of projective geometry, motion parallax, depth cues, and grey-scale data to give the impression of a realistic third dimension. These display techniques can be applied to any object whose internal structure can be measured or computed. Medical applications are of particular interest because $2\frac{1}{2}$-D display

[2] A "voxel" is a volume element in the way that a "pixel" is a picture element.

techniques have already proven effective in such areas as radiation therapy and noninvasive surgical planning.

The Voxel Processor architecture has substantial advantages for medical applications when compared to most other approaches. It is capable of over 500 times the speed of the fastest software-based techniques and 20 times the speed of competing hardware technology. It requires minimal preprocessing and data preparation; has the potential for true real-time display, manipulation, and editing of medical objects regardless of their complexity; is modular and expandable; and can be realized inexpensively using VLSI technology.

The architecture of the Voxel Processor is under development by a team supervised by Sam Goldwasser and consisting of Anthony Reynolds (Reynolds, 1985a) (basic algorithms); Ed Walsh (engineering team leader); Larry Ashery, and Andrew Wolff (hardware implementation and prototype); Ted Bapty (system integration and enhancements); David Baraff (interface software); Chris Warren (64-cube simulation); and John Summers (enhanced software and shading hardware). The team was responsible for the prototype system (described later), which has been operational for over a year and is routinely used for experimentation and support of clinical studies and other imaging research projects.

Real-Time 3-D Physician's Workstation

An interactive physician's workstation has been developed that provides many desired display capabilities with real-time update rates (Goldwasser *et al.* 1985b). The workstation is centered around an implementation of the Voxel Processor architecture known as the Voxel Processor Prototype (VPP). The VPP provides for the true real-time display and manipulation of a small (64-cube) volume of a 3-D dataset. However, the architecture is expandable so that a system capable of accommodating a larger dataset with full spatial and density resolution can be realized with no degradation of real-time performance. Capabilities implemented in the VPP include real-time rotation of shaded binary or grey-scale 3-D objects, interactive segmentation based on tissue density, instantaneous reslicing of the original data, and the four-dimensional (4-D) display of dynamically changing objects.

The workstation consists of four components: a general-purpose host computer for system control, data preprocessing, high-resolution image generation, and file management; a 3-D VPP for interaction with 3-D objects; a 2-D frame buffer for display of images and graphics; and an interactive control panel (ICP).

The software environment is hierarchically organized and consists of four major components:

- Preprocessing utilities provide efficient conversion and interpolation of the 2-D image data, filtering, and image database management.
- Volume-of-interest extraction interactively enables the rapid location, extraction, and, if necessary, subsampling of a volume of interest for display on the VPP.
- VPP interactive control implements the real-time control of the VPP using the ICP.
- Offline image generation produces images of full-size medical datasets (up to 512 x 512 x 512) using optimized software running on the host VAX-11/750. The time for each image depends upon dataset size but is typically one to two minutes.

Enhanced Voxel Processor Hardware. Several designs for expanded Voxel Processor systems are being developed, and one of these is being implemented. The alternatives are differentiated by trade-offs between speed, complexity, and flexibility.

The system currently under construction includes:

Large Object Memory: The Mini Voxel System (MVS) will support an object memory of up to 16 million eight-bit voxels. This will permit full-size medical datasets (typically 256 x 256 x 64) to be resident in the system and instantaneously accessible. The memory will be reconfigurable to accommodate arbitrary object shapes.

Interpolation: The system will incorporate on-the-fly interpolation to generate cubical voxels, eliminating this as a preprocessing step.

Shading and Video Output: A flexible video processor will provide a variety of shading and pseudocolor options. Eventually, full gradient shading hardware will be added.

Microcoded System: The MVS will be fully programmable from the host processor and will be capable of performing some of the tasks previously assigned to the host computer.

Image Space Gradient Shading. A hardware design and software simulation of a gradient shading postprocessor for the Voxel Processor has been implemented. Currently, the gradient shading simulator is being tested with grey-scale data and with output from ray-tracing image-generation algorithms developed by Patrick Decuypere and Ravi Gururaj. John Summers' M.S.E. thesis (Summers, 1986) addresses the algorithm analysis and hardware design of a real-time implementation of the gradient shading technique.

Interpolation of CT Data. Andrzej Zawodniak is investigating nonlinear interpolation methods that can be used for CT data. A program to compare linear interpolation to cubic spline and Aitken's interpolation algorithms has been developed. This same program is being tested by selecting one point from any displayed CT data slice for the purpose of interpolating all relevant neighboring slices with the study sample.

Ray-Tracing Test Bed. Patrick Decuypere is developing a set of simulations of advanced ray-tracing-based volume rendering. These incorporate trilinear interpolation in object space and gradient shading in image space. Tests on medical datasets have resulted in some of the best shaded graphics images ever produced. The system supports datasets of arbitrary size and includes perspective capability in addition to orthographic projection. One of the significant aspects of this work is that all the techniques being developed are directly applicable to high-speed hardware implementation.

Related Research

Several CIS faculty members are carrying on research projects that are not directly associated with any of the four main laboratories.

Peter Buneman is working with graduate student Atsushi Ohori on connections between inheritance, data types, and databases. Motivated by the belief that a strong type system is essential for large-scale program development, they are developing a formal model of inheritance in order to assign data types to common database operations and to extend these to capture operations in knowledge bases and object-oriented languages. Moreover, they believe it is important to have a formal model of types that allows most types to be inferred rather than explicitly declared by the programmer and that provides a better understanding of the semantics of database construction. In particular, Ohori has assigned a formal semantics to relational database theory, and is studying a denotational semantics that provides a representation of relational database operations and constraints such as functional dependencies.

Susan Davidson is studying distributed databases—collections of databases that are (often) physically separated yet logically form a single database. She is particularly interested in the fault tolerance of these systems. Currently, she is looking at partition failures, that is, failures that isolate some subset of the nodes in the system. Algorithms for handling partition failure have been developed, and Davidson is in the process of analyzing these algorithms, using both combinatoric and simulation techniques. She has also studied agreement algorithms in this scenario.

Jean Gallier and graduate student Stan Raatz are investigating several issues in logic programming. One project concerns the design of an interpreter for a class of logic programs that allow negative Horn clauses and certain kinds of negative goals. A graph-based rather than a stack-based interpreter has been implemented because the graph method is more efficient (because subgoals are computed only once) and makes it possible to detect and prevent certain kinds of loops common in Prolog programs. Gallier and Raatz are also studying methods of incorporating equality into logic programming languages. This would make it possible to write function definitions in a logic programming framework, thus combining the features of functional programming languages with those of logic programming languages.

Eva Ma is studying three aspects of reconfigurable computer architectures: theoretical frameworks for reconfigurable systems, special-purpose reconfigurable computers, and dynamic computer networks. The theoretical work centers on developing analytical models for such systems, models for selecting optimal reconfiguration strategies, and protocols for accomplishing reconfiguration. Work on special-purpose reconfigurable computers involves development of a methodology for the design of architecture features based on an analysis of application characteristics (such as the computation and communication requirements of the application program). In the area of dynamic networks, algorithms are being designed to analyze the impact of topological variations (which can occur due to node or link failures, relocation of nodes, or network expansion) on network reliability and performance. Efficient algorithms are critical in supporting network reconfiguration to adapt to these topological variations.

Michael Palis, a new member of the CIS faculty, is working with graduate students Sunil Shende and David Wei on parallel algorithms for language parsing. Although there are many results known for parallel language recognition, little work has been done on parallel parsing (building a parse tree for an input string, rather than just deciding if the input string is in the language generated by a particular grammar). Fast (polylog time) parallel algorithms exist for language recognition, but they invariably require an impractical number of processors. Palis and Shende are trying to develop algorithms for parallel language parsing that are efficient in terms of both time and processor requirements. For this work, a hypothetical (parallel random access machine) computer is being used as the model of computation. Using a more practical model of computation, Palis and Wei are studying techniques for implementing parallel parsing of tree-adjoining grammars (see Studies on Grammatical Formalisms) on systolic arrays. Other problems they are currently investigating include parallel algorithms for syntax-directed translation, text editing, and sequence comparisons.

Scott Weinstein, a member of the philosophy department, is working with Daniel Osherson (Cognitive and Brain Sciences, MIT) and Michael Stob (Mathematics and Computer Science, Calvin College) on machine inductive inference. This research has both a scientific and a technological import. On the scientific side, Osherson, Stob, and Weinstein, along with several other investigators in theoretical computer science (including D. Angluin at Yale, J. Case at SUNY Buffalo, and L. Valiant at Harvard), are articulating a mathematical theory that provides a framework for studying problems of inductive inference and learning in a variety of domains. The idea is to provide a sound theoretical basis for the application of machine inductive inference to such problems as language acquisition. In the technological domain, the work is geared toward providing a basis for applications to problems of inductive inference of a quite general character, such as automatic program synthesis (a survey of this work is contained in Osherson, Stob & Weinstein, 1986).

The Cognitive Science Program

In the spring of 1978, the University of Pennsylvania hosted two workshops, one on theoretical aspects of discourse processing and the other on language acquisition; in retrospect these workshops can be viewed as having launched the university's cognitive science program. These workshops brought together researchers in computer science, linguistics, philosophy, and psychology to discuss scientific problems whose solutions (and even proper formulations) could not be achieved within the framework of any of these disciplines alone. The workshops were successful in initiating discussions among the university faculty who attended, and during the following academic year, a seminar was established to continue these interactions. As the seminar developed, it became clear that a series of tutorials would be needed if workers in such diverse areas were to be able to embark on joint research projects—a prospect that seemed at the time the only way of attacking the important scientific problems that had been brought to light. It was also realized that such tutorials could be a valuable vehicle for educating new researchers, so the seminars were expanded to include a limited number of graduate students as well as faculty from the relevant disciplines.

Since that time, the seminars have continued weekly and are one of the major activities of the university's program in cognitive science. The seminars have grown to include tutorial presentations designed to help researchers in different fields understand one another's work and to provide a forum for the discussion of research in progress by members of the university's cognitive science community and invited speakers. Recent visitors have spoken on the connection between complexity theory and problems in natural language processing, developments in the semantics of polymorphically typed programming languages and their bearing on natural language semantics, recent results of psycholinguistic studies relevant to the structure of the lexicon, and philosophical and computational aspects of machine-inductive inference.

The cognitive science program has already had a significant impact on research and education at the university. The weekly cognitive science seminars have spawned several other seminars on specialized topics (such as visual perception, natural language processing, and theoretical aspects of discourse interpretation and language acquisition) that have in turn fostered joint research projects undertaken by faculty and graduate students across the range of departments represented in the program. The program has recently begun to have an effect on undergraduate education as well, with the initiation of a dual-degree program, the focus of which is computer and cognitive science, between the School of Engineering and Applied Science and the School of Arts and Sciences.

Research Funding

The AI research reported here is supported by grants from the National Science Foundation (Coordinated Experimental Research and Intelligent Systems), the Army Research Office (ARO), the Defense Advanced Research Projects Agency (DARPA), the Office of Naval Research (ONR), the Air Force Office of Scientific Research (AFOSR), NASA, the Alfred P. Sloan Foundation, Burroughs-SDC, Digital Equipment Corporation, Hewlett-Packard, IBM, the LORD Corporation, and RCA.

References

Allen, James F. 1982. *Recognizing Intentions from Natural Language Utterances.* Cambridge, Massachusetts: MIT Press, .

Badler, N., J. O'Rourke, & H. Toltzis, 1979. "A Spherical Representation of a Human Body for Visualizing Movement." In *Proceedings of the IEEE.* Silver Spring, Maryland: IEEE, 1397–1403.

Badler, N. I. 1986. *A Representation for Natural Human Movement.* Tech. Rep. MS-CIS-86-23, Department of Computer and Information Science, University of Pennsylvania.

Badler, N. I. & R. Bajcsy, 1978. "Three-Dimensional Representations for Computer Graphics and Computer Vision." *Computer Graphics*, 12: 153–160.

Badler, N. I., J. D. Korein, J. U. Korein, G.M. Radack, & L. S. Brotman, 1986. "Positioning and Animating Human Figures in a Task-Oriented Environment." *The Visual Computer*, 1: 212–220.

Bernstein, A. & P. K. Harter Jr., 1981. "Proving Real-Time Properties of Programs with Temporal Logic." In *Proceedings of the 1981 Symposium on Operating System Principles*, 1–11.

Brachman, R. J. 1978. *A Structural Paradigm for Representing Knowledge.* Tech. Rep. 3605. Cambridge, Massachusettes: Bolt Beranek and Newman, Inc., .

Brachman, R. J. & J. G. Schmolze, 1985. "An Overview of the KL-ONE Knowledge Representation System." *Cognitive Science*, 9: 171–216.

Cheikes, B. A. 1985. *Monitor Offers on a Dynamic Database: The Search for Relevance.* Tech. Rep. MS-CIS-85-43, Department of Computer and Information Science, University of Pennsylvania.

Cohen, P. 1978. *On Knowing What to Say: Planning Speech Acts.* Tech. Rep. 118, Department of Computer Science, University of Toronto.

Felty, A. 1986. "An Integration of Resolution and Natural Deduction Theorem Proving." In *Proceedings of AAAI-86.* Menlo Park, California: American Association for Artificial Intelligence.

Finin, T., B. L. Webber, & Joshi, A. K. 1986. *Natural Language Interactions with Artificial Experts.* Tech. Rep. MS-CIS-86-16, Department of Computer and Information Science, University of Pennsylvania.

Fishwick, P. A. 1986. "Hierarchical Reasoning: Simulating Complex Processes over Multiple Levels of Abstraction". PhD diss., Department of Computer and Information Science, University of Pennsylvania.

Gangel, J. S. 1985. *A Motion Verb Interface to a Task Animation System.* Master's thesis, Department of Computer and Information Science, University of Pennsylvania.

Goldwasser, S. 1984. *GRASP News: Quarterly Progress Report.* Tech. Rep. MS-CIS-85-11, Department of Computer and Information Science, University of Pennsylvania.

Goldwasser, S., G. Hager, & F. Solina, 1986. *GRASP News: Quarterly Progress Report.* Tech. Rep. Department of Computer and Information Science, University of Pennsylvania.

Goldwasser, S. & J. McCarthy, 1985a. *GRASP News: Quarterly Progress Report.* Tech. Rep., Department of Computer and Information Science, University of Pennsylvania.

Goldwasser, S. & J. McCarthy, 1985b. *GRASP News: Quarterly Progress Report.* Tech. Rep., Department of Computer and Information Science, University of Pennsylvania.

Goldwasser, S. M. 1985. "The VOXEL Processor Architecture for Real-Time Display and Manipulation of 3-D Medical Objects." In *Proceedings of the 6th Annual Conference and Exposition of the NCGA.* Fairfax, Virginia: National Computer Graphics Association, 71–80.

Goldwasser, S. M. & R. A. Reynolds, 1983. "Architecture for the Display and Manipulation of Three-Dimensional Objects." In *Proceedings of the 1983 International Conference on Parallel Processing.* Bellaire Michigan, 269–274.

Goldwasser, S. M., R. A. Reynolds, T. Bapty, D. Baraff, J. Summers, D. Talton, & E. Walsh, 1985. "A 3-D Physician's Workstation with Real-Time Performance." *IEEE Computer Graphics and Applications*, 5(12): 44–57.

Goldwasser, S. M., R. A. Reynolds, D.Talton, D., & Walsh, E. 1986. "Real-Time Display and Manipulation of 3-D CT, PET, and NMR Data." Paper presented at the International Workshop on Physics and Engineering in Computerized Multidimensional Imaging and Processing, April.

Granor, T. E. 1986. "A User Interface Management System Generator." PhD diss., Department of Computer and Information Science, University of Pennsylvania.

Grice, H. P. 1975. "Logic and Conversation." In *Syntax and Semantics*, New York: Academic Press.

Grosz, B., A. K. Joshi, & S. Weinstein, 1986. "Towards a Computational Theory of Discourse Interpretation." *Journal of Computational Linguistics.* (forthcoming).

Hager, G. 1985. *Computational Aspects of Proofs in Modal Logic.* Master's thesis, Department of Computer and Information Science, University of Pennsylvania.

Hirschberg, J. B. 1985. "A Theory of Scalar Implicature". PhD diss., Department of Computer and Information Science, University of Pennsylvania.

Hoare, C. A. R. 1985. *Communicating Sequential Processes.* Engelwood Cliffs, New Jersey: Prentice Hall, .

Izaguirre, A. 1986. "Automatic Methods of Generating the Dynamics Equations Using LISP." In *Proceedings of the IEEE Conference on Robotics and Automation, San Francisco.* Silver Spring, Maryland: IEEE, 220–226.

Joshi, A. K. 1982. "Mutual Beliefs in Question Answering Systems." In N. Smith, ed., *Mutual Belief.* New York: Academic Press, .

Joshi, A. K., L. S. Levy, & M. Takahashi, 1975. "Tree Adjunct Grammars." *Journal of Computer and System Sciences*, 10(1).

Joshi, A. K., B. Webber, & R. Weischedel, 1984. "Living Up to Expectations: Computing Expert Responses." In *Proceedings of AAAI-84, Austin, Texas.* Menlo Park, California: American Association for Artificial Intelligence.

Joshi, A. K., D. Webber, & R. Weischedel, 1984. "Preventing False Inferences." In *Proceedings of COLING-84, Stanford, California.*

Kamp, H. 1984. "A Theory of Truth and Semantic Representation." In J. Groenendijk, T. Janssen, and M. Stokhof, ed.s, *Truth, Interpretation and Information.* Cinnaminson, NJ: Foris Publications, 1–41

Kaplan, J. 1982. "Cooperative Responses from a Portable Natural Language Database Query System." In M. Brady, ed., *Computational Models of Discourse*, Cambridge, Massachusetts: MIT Press, . .

Kass, R., R. Katriel, & T. Finin, 1986. *Breaking the Primitive Concept Barrier.* Tech. Rep. MS-CIS-86-36, Department of Computer and Information Science, University of Pennsylvania.

Korein, J. U. 1985. "A Geometric Investigation of Reach". PhD diss., Department of Computer and Information Science, University of Pennsylvania.

Kroch, A. & A. K. Joshi, 1985. *Linguistic Relevance of Tree Adjoining Grammars.* Tech. Rep. MS-CIS-85-18, Department of Computer and Information Science, University of Pennsylvania.

Krotkov, E. & J. P. Martin, 1986. *Range from Focus.* Tech. Rep. MS-CIS-86-09, Department of Computer and Information Science, University of Pennsylvania.

Krotkov, E., J. F. Summers, & A. F. Fuma, 1986. *The Pennsylvania Active Camera System.* Tech. Rep. MS-CIS-86-15, Department of Computer and Information Science, University of Pennsylvania.

Mays, E. 1980. "Failures in Natural Language Systems: Application to Data Base Query Systems." In *Proceedings of the AAAI-80, Stanford, California.* Menlo Park, California: American Association for Artificial Intelligence.

Mays, E. 1984. "A Modal Temporal Logic For Reasoning About Changing Databases With Applications To Natural Language Question Answering". PhD diss., Department of Computer and Information Science, University of Pennsylvania.

McCoy, K. F. 1985. "Correcting Object-Related Misconceptions". PhD diss., Department of Computer and Information Science, University of Pennsylvania.

McDonald, D. 1983. "Dependency Directed Control: Its Implications for Natural Language Generation." In N. Cercone, ed., *Computational Linguistics*, Elmsford, New York: Pergamon Press.

McKeown, K. R. 1985. "Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text". PhD diss., Department of Computer and Information Science, University of Pennsylvania.

Nadathur, G. & D. Miller, 1986. "Higher-Order Logic Programming." In *Third International Logic Programming Conference.* London: Imperial College.

Nadathur, G. & Miller, D. 1986. "Some Uses of Higher-Order Logic in Computational Linguistics." In *Proceedings of the 24th Meeting of the Association for Computational Linguistics.* New York: Association for Computational Linguistics 247–256, ACL, New York, June.

Osherson, D. N., M. Stob, & S. Weinstein, 1986. *Systems That Learn: An Introduction to Learning Theory for Cognitive and Computer Scientists.* Cambridge, Massachusetts: MIT Press, .

Platt, S. 1985. *A Structural Model of the Human Face.* PhD diss., Department of Computer and Information Science, University of Pennsylvania.

Pollack, M. E. 1985. "Information Sought and Information Provided." In *Proceedings of CHI'85.* New York: Association for Computing Machinery, 155–160.

Pollack, M. E. 1986. "A Model of Plan Inference that Distinguishes Between Actor's and Observer's Beliefs." In *Proceedings of the 24th Meeting of the Association for Computational Linguistics*, ACL, New York, June.

Pollard, C. 1984. *Generalized Phrase Structure Grammars, Head Grammars and Natural Language.* PhD diss., Department of Linguistics, Stanford University.

Reynolds, R. A. 1985. *Fast Methods for 3-D Display of Medical Objects.* PhD diss., Department of Computer and Information Science, University of Pennsylvania.

R. Rubinoff. 1986. *Adapting MUMBLE: Experience in Natural Language Generation.* Tech. Rep. MS-CIS-86-32, Department of Computer and Information Science, University of Pennsylvania.

R. Rubinoff. 1985. Explaining Concepts in Expert Systems: the CLEAR System. In *Proceedings of the 2nd Conference on Artificial Intelligence Applications*, IEEE, Miami, December.

Shastri, L. 1985. *Evidential Reasoning in Semantic Networks: A Formal Theory and Its Parallel Implementation.* PhD diss., Department of Computer Science, University of Rochester.

Shastri, L. 1985. A Massively Parallel Encoding of Semantic Networks. In *Proceedings of the Distributed Artificial Intelligence Workshop*, AAAI, Sea Ranch CA, December.

Shastri, L & J A Feldman, 1985. "Evidential Reasoning in Semantic Networks: A Formal Theoiy." In *Proceedings of IJCAI-85*, 465–474, IJCAI, Los Angeles, August.

Smitley, D., S. M. Goldwasser, & I. Lee, 1985 *IPON-Advanced Architectural Framework for Image Processing* Tech Rep MS-CIS-85-13, Department of Computei and Information Science, University of Pennsylvania.

Stansfield, S 1986. "Primitives, Features, and Exploratory Procedures: Building a Robot Tactile Perception System." In *Proceedings of the IEEE Conference on Robotics and Automation*, 1274–1279, IEEE, San Francisco, April.

Stansfield, S. 1985 "Representation and Contiol Within an Intelligent, Active, Multisensor system for Object Recognition." In *Proceedings of the Conference on Systems, Man, and Cybernetics*, 215–219, IEEE, Tucson, November

Summers, J. 1986. *The Design of Real-Time Gradient Shading for Voxel-Based Objects* Master's thesis, Department of Computer and Information Science, University of Pennsylvania.

Tiinkle, J C 1985. *Frictionless Grasping.* Tech Rep. MS-CIS-85-46,

Trinkle, J. C 1986 "Planning Frictionless, Self-Adjusting, Enveloping Grasps " In *Proceedings of the Japan-USA Joint Symposium on Flexible Automation*, Osaka, July.

Vijay-Shanker, K & A. K Joshi, 1985. "Some Computational Properties of Tree Adjoining Grammars." In *Proceedings of the 23rd Meeting of the Association for Computational Linguistics*, 82–93, ACL, July

Vijay-Shanker, K , D. J. Weir, & A. K. Joshi, 1986. "Tiee Adjoining and Head Wrapping " In *Proceedings of COLING-86*, Bonn, August

Webber, B. 1982 "So What Can We Talk About Now." In M Brady, ed , *Computational Models of Discourse* Cambridge, Massachusetts: MIT Press, .

Weir, D J., K. Vijay-Shanker, & A. K. Joshi. 1986. "The Relationship Between Tree Adjoining Grammars and Head Grammars." In *Proceedings of the 24th Meeting of the Association for Computational Linguistics*, 67–74, ACL, New York, June

Zhang, H & R Paul 1986. "The Design of a Robot Force/Motion Server " In *Proceedings of the IEEE Conference on Robotics and Automation*, New York: IEEE, 1878–1883

Zwarico, A. & I. Lee 1985. "Proving a Network of Real-Time Processes Correct " In *Proceedings of the Real-Time Systems Symposium*, 169–177, December