SUMMARY OF PROGRESS IN IMAGE UNDERSTANDING RESEARCH AT THE UNIVERSITY OF MASSACHUSETTS

Allen R. Hanson
Edward M. Riseman

COINS Technical Report 87-20

March 1987

This work was supported by DARPA under Contracts N00014-82-K-0464, DACA76-85-C-0008, and DACA76-86-C-0015, by AFOSR under Contracts F49620-86-C-0041 and 86-0021, by DMA under Contract 800-85-C-0012, and by the National Science Foundation under grants DCR-8318776, and DCR-8500332.

Summary of Progress in Image Understanding at the University of Massachusetts

Allen R. Hanson and Edward M. Riseman

Computer and Information Science Department
University of Massachusetts at Amherst

ABSTRACT

Image Understanding research at the University of Massachusetts encompasses a range of research, most of which is directed towards the integration of a diverse set of processes to achieve a general real-time knowledge-based interpretation system. In particular we are concentrating on integrating projects involving object identification in static images, depth recovery from motion analysis, a real-time parallel architecture, and mobile vehicle navigation. This system will be applied to a variety of task domains of natural scenes including road scenes and aerial images, and will also be used to control a mobile robot moving through both known and unknown outdoor domains.

This summary documents several areas of research at the University of Massachusetts that are entirely or partially supported under the DARPA image understanding program. The work, much of which is documented in papers in these proceedings, is divided into several areas:

1. Knowledge-Based Vision
2. Perceptual Organization (intermediate processing)
3. 3D Models, Matching, and Surface Recovery
4. Mobile Robot Navigation
5. Image Understanding Architecture
6. Motion Analysis
7. Low-Level Vision

This work was supported by DARPA under Contracts N00014-82-K-0464, DACA76-85-C-0008, and DACA76-86-C-0015, by AFOSR under contract F49620-86-C-0041 and 86-0021, by DMA under contract 800-85-C-0012 and by the National Science Foundation under grants DCR-8318776, and DCR-8500332.

1 Knowledge-Based Vision

A central problem in image understanding is the representation and use of all available sources of domain knowledge during the interpretation process. Each of the many different kinds of knowledge that may be relevant during the interpretation process imposes different kinds of constraints on the underlying representation and may lead to very different kinds of strategies for its effective use. Over the past several years, we have developed the notion of a 'schema' as the basic unit of knowledge representation in the VISIONS system. Within the schema system image interpretation is the process of instantiating a subset of schemas to build a description of the three-dimensional scene which gave rise to the image. Knowledge is represented in an abstraction hierarchy of schema nodes by part/subpart descriptions, class/subclass descriptions, and expected relationships between schemas; the resultant hierarchical graph constitutes the VISIONS knowledge network. This work has evolved for a long period of time, with recent work documented in [19,23,45].

Each schema node may be viewed as a 'packet' of information related to the object being described, including properties and relations of extracted image events as well as control information expressed in the form of interpretation strategies. One or more of these strategies are executed when a schema is instantiated (i.e. when a copy is activated), to process a specific area of the image. Schema activation may be either bottom-up, where image descriptions imply the potential relevance of a schema, or top-down as the result of the context of a partial interpretation written on a blackboard communication structure by other schemas. Many schemas may be active at any one time, and the interpretation strategies provide control over the parallel interpretation processes and make use of a set of object-independent processes called knowledge sources. In our system schemas communicate indirectly by posting object hypotheses on the blackboard.

The system is organized around three levels of data representation and types of processing. At the low-level, the representations are in the form of numerical arrays of sensory data with processes for extracting the image events that will form the intermediate representation. At the intermediate level, the representation is composed of symbolic tokens representing regions, lines, surfaces and the attributes of these primitive elements (which might include local motion and depth information). The intermediate representation is stored in a data base called the intermediate symbolic representation (ISR) which supports grouping (perceptual organization) and information fusion processes that are employed to develop aggregations of existing tokens to form new tokens. At the high level, the representation is a set of object hypotheses and active schema instances which control the intermediate and low-level processes. Control initially proceeds in a data-directed manner and later is significantly top-down in a knowledge-directed manner.

Based on our experience with an initial implementation of the schema system and a set of experiments designed to interpret reasonably complex house scenes [23,24,45], a new schema system and support environment has been designed and partially implemented [19]. Two new tools, the Intermediate Symbolic Representation and the Schema Shell, have been developed and are currently being tested and extended using the interpretation of road scenes as a second experimental task domain. A third experimental domain of aerial image analysis for cartography applications is planned for the near future.

The input to high-level vision processes is *intermediate-level* data, which is the output from low-level processes such as line extraction and region segmentation, and of intermediate-level processes of grouping and selection. In our environment, intermediate level image descriptions are stored in the Intermediate Symbolic Representation, (or ISR). The ISR is a database which has been custom-built for the efficient storage, manipulation, and retrieval of abstract image data. The fundamental unit of representation is the *token*, each of which has a unique name, and a list of feature slots. The ISR can be used to store anything that can be fully characterized by a list of features and values; some of the image events currently stored include region segments, extracted edge lines, areas of homogeneous texture, rectilinear line groups, and region-line relations. The benefits which result from imposing a uniform representation and user interface on all intermediate level tokens are enormous. It now becomes natural to think in terms of multistage and hierarchical grouping processes which take in tokens at one level of abstraction and produce tokens at the next higher level [39]. It also becomes more tractable to compare different types of tokens, which is necessary, for example, when relating edge lines to the regions they intersect [6]. Of course, the sharing of results and the elimination of data reformatting routines are obvious advantages.

At the highest level, tokens are partitioned according to the *image* they were extracted from. There is also an intermediate level of partitioning called the *tokenset*. Each feature associated with the tokens in a tokenset has a name, a value slot, a data type, and a computation function. Since most tokens have a physical realization in an image, a special bitplane data type is provided for representing the subset of image pixels which are associated with a token. Operators exist for taking the intersection, union, and difference of token bitplanes.

The ISR supports efficient access functions to tokens and sets of tokens; some of these access functions are associative in nature in that tokens may be accessed by means of constraints on feature values. In addition, the features may be precomputed or computed on a demand basis when the token/feature slot is accessed. The ISR allows a schema to create a token during an interpretation, create its bitplane either from scratch or as some combination of token bitplanes, and access its features, at which time new feature values will be automatically calculated for the new token. Thus, it is possible for schemas to dynamically "correct" misleading segmentations based on combinations of top-down knowledge.

The Schema Shell is an environment tool that supports the development of large systems of schemas. Each schema contains knowledge about recognizing a class of objects. It has data declarations for collecting relevant information, and procedures for determining whether, when and how to ascertain that information. Parameterized instances of schemas are then invoked to interpret an image. The Schema Shell provides mechanisms for building schemas and simulates a distributed environment (until parallel hardware arrives) in which an arbitrary number of schema instances may run concurrently. Schema instances communicate through a central blackboard. At any point during its processing a schema strategy may write an arbitrary message to the blackboard. Every other schema is then free to read, erase or modify that message. This provides for a single, uniform communication mechanism between schemas which can also be easily implemented on a variety of distributed architectures.

Bottom-up activation of schemas can be accomplished by forming initial object hypotheses on the basis of attributes of the initial image description expressed in terms of lines and regions. Previously we have reported on rule-based approaches to initial hypothesis generation [6,24] which used a heuristic approach to forming constraints (rules) based on a theoretical Bayesian approach to maximum likelihood decisions over feature distributions. Recently, Lehrer and Reynolds [33] have extended the work and have developed a new object hypothesis system based on the Shafer-Dempster [17,40] theory of evidence. Their approach provides a more formal and theoretical foundation for the definition and interpretation of world knowledge. Object specific knowledge is defined automatically using statistical information obtained from a set of training object instances and a computationally efficient approach to the Dempster-Shafer theory of evidence is used for the representation and combination of evidence from multiple sources.

In this approach, the relationship between an object and its attributes is captured in a "plausibility" function. When applied to the primitive tokens (e.g. regions) the plausibility functions return evidence for or against an object hypothesis. The evidence from multiple plausibility functions is combined using an efficient computational algorithm to produce the final hypothesis. A large scale experiment is being planned for comparing and evaluating the results of the two systems.

2 Perceptual Organization (Intermediate Processing)

We are initially viewing the task of perceptual organization and grouping as the extraction of relevant structure from overfragmented and incomplete descriptions and the construction of more abstract descriptions from less abstract ones. By this we mean algorithms which have as input the tokens produced by the low-level system and other grouping operations (region, lines, flow fields,...) and have as output more complex tokens generated by grouping strategies based on the *relations* between the tokens. The goal of this type of 'intermediate' level processing is the reduction of the substantial representational gap which exists between the low level image descriptors and the primitives with which the high level semantic descriptions are constructed. The process of abstraction thus involves the search for events which can be more concisely described as a unit and which results in a description which may be more relevant to the evolving semantic interpretation.

Over the past two years some progress has been made in developing grouping algorithms at the intermediate level of representation. The intermediate symbolic representation, described briefly earlier, has been developed as the supporting representation for this work and several algorithms developed previously have been cast within this framework. A number of the local strategies for using the rank-ordered object hypotheses generated by the rule-structured initial object hypothesis system [24] can be viewed as grouping strategies. The extensions to this system developed by Belknap [5], which fuses information across multiple token types by means of relations expressed as rules, is also a form of grouping and has successfully generated object hypotheses from a combination of geometric and spectral features. Boldt [11] has developed a scale-sensitive hierarchical algorithm for grouping collinear line segments into progressively longer segments on the basis of geometric properties of the hypothesized group as well as the similarity of image features along both sides of the component lines. A summary of these algorithms and a more comprehensive discussion of their relationship to perceptual organization and grouping may be found in [23].

As a result of these preliminary studies related to grouping, we [11,39] are developing a computational framework for geometric grouping and other organizational algorithms which addresses a set of overlapping issues. Clearly one must consider the extraction and representation of primitive tokens, the features of these tokens, and important relations between the tokens. In the case of geometric grouping algorithms this would include the extraction of lines and geometric relations such as collinearity, parallelness, relative angle, and spatial proximity derived from the Gestalt Laws of perceptual organization. One must also provide the means for expressing domain constraints in terms of these relations; i.e. grouping *strategies* must be defined and invoked based on knowledge of the domain and the current state of the system. Finally the system must deal explicitly with the problem of search, and its relation to the objects in the domain which are to be hypothesized and identified. In general each step of any grouping strategy must apply constraints which either significantly reduce the search space and/or add important information to the descriptive power of the system.

A number of algorithms are being developed at UMASS which satisfy these requirements and a computational framework has been proposed for confronting the issues described above. We view the grouping and search processes as part of a four-stage iterative grouping and extraction strategy which can be summarized as follows:

- *Primitive Structure Generation*: These processes provide the primitives (regions, lines, possibly surfaces, and in general, tokens) which are the input to the grouping and hypothesis generation process described next.
- *Linked Structure Generation*: This step applies very general geometric constraints to obtain graphs within which search processes can be applied to identify specific objects of interest. For example rectilinear structures which would contain rectangles or other simple geometric structures. This is essential for generating search spaces of reasonable size.
- *Subgraph Extraction*: This step involves the extraction of specific structures "one step up" the abstraction hierarchy, and uses the linked structures to constrain the search.
- *Replacement and Iteration*: Having extracted more abstract tokens, these can now play the role of primitives in another pass of grouping and extraction.

In [11,12] this strategy has been applied with striking results for the purpose of extracting straight lines. In [39] this strategy is being applied for the purpose of extracting rectilinear structures. In unpublished work Lance Williams is developing an algorithm for using a flow field generated from a motion sequence of images, to assist in the straight line extraction and temporal grouping process with excellent preliminary results.

While many of the grouping algorithms discussed above are designed to be applied uniformly across an image, many of them are computationally intensive. In addition, it often does not make sense to apply them in a uniform fashion because they may not be applicable to all portions of the image. For example, the rectilinear grouping algorithm [39] probably should not be applied in heavily textured areas. Consequently, we are examining strategies in which the algorithms are applied selectively to those areas of the image for which they are most suited. Kohl [29] has been developing a schema-based system called GOLDIE for intelligently controlling the application of parameterized low- and intermediate-level processes on the basis of goals and constraints generated by the high-level interpretation system. Initially, GOLDIE (for Goal-Directed Intermediate Level Executive) was formulated as a goal-oriented resegmentation system which allowed top-down control over the low-level level segmentation processes and this remains an important aspect of its function. However, it also has become clear that top-down control of the intermediate-level grouping processes is required; consequently GOLDIE has been extended to include these processes in its repertoire. Both the Boldt line grouping algorithm and a rule-based region merging algorithm are incorporated into it and we are examining further extensions. GOLDIE responds to requests from the interpretation processes through the goal structure by translating the goals into appropriate low- and intermediate-level process specifications and then executing the process. The constraints imposed on the output of the process can be quite general; if the resulting structure does not satisfy the request, the system attempts to generate other strategies, using whatever contextual and semantic knowledge is available, in order to meet the constraints.

3 3D Models, Matching and Surface Recovery

There have been two recent research efforts in our group directed towards 3D object recognition and surface recovery. Two-dimensional images provide us with cues to the three-dimensional structure of objects which can be used for recognition or description. We are exploring a methodology of generic (characteristic) views for model-based recognition. The primary feature which characterizes each of the generic views is the binary relationships between pairs of lines which are visible in the same view. For the problem of reconstructing surfaces we have adopted an approach of constructing the envelope of the object from changes of the contours under planar motion of the camera. What these two approaches have in common is that they both use geometric knowledge about contours. Our efforts are presented in a bit more detail below.

Often small changes in the viewpoint will only produce small changes in the appearance of an object. If we measure the visibility of features, (e.g. whether or not an edge or vertex is visible), they will be stable over a wide range of viewpoints. Such a set of viewpoints of an object is called a generic view. The model of an object consists of all of its generic views. The classification of the types of features and transitions for smooth surfaces has been analyzed [15,26,28] and we have extended these results to piecewise smooth surfaces [16]. Piecewise smooth surfaces are made up of patches of smooth surfaces which meet at creases. This type of surface subsumes both polyhedra and smooth surfaces. If the pieces are planar, then the surface one gets is polyhedral. If there are no creases, then the surface is smooth.

Using this approach Kitchen and Burns are constructing a 3D modelbase of objects, which will be analysed in order to make predictions about the visual configurations that views of the objects will give rise to in an image. These predictions are being organized into a hierarchical structure with explicit sharing of predictions common to multiple views. At recognition time, extracted image features are to be matched against this hierarchy in order to quickly establish what view of what object is seen, even if there are many possible objects in the modelbase. Once the view is known and the correspondence determined between image features and 3D object parts, it is possible to solve numerically for the object's pose parameters, using general methods or view-specific methods where advantageous [25,36,37].

We are proceeding with an implementation and analysis of this approach as applied to recognizing rigid polyhedra. Currently a system for modeling polygonal prisms has been implemented, along with a graphics interface as a tool for exploring the geometry of predictions. More important, we have an initial system implemented for making predictions about the appearances of these prisms and organizing them into a hierarchy for recognition purposes. Work is also in progress on robust and efficient techniques for solving for object pose which are tailored for specific classes of views.

In a separate effort, Giblin and Weiss have mathematically analyzed the reconstruction of surfaces from profiles and have derived an algorithm to implement it. Information can be derived about the shape of an object from a single profile, and with multiple views the shape can often be determined uniquely. Based on this analysis they have found an algorithm which can be used to produce a depth map of the surface. However, for some applications it may not be necessary to produce a depth map at all (for example in recognition problems), and thus they have also provided an algorithm which computes Gauss and mean curvatures without first computing the depth map. It should be noted that the Gauss and mean curvatures have been used by other researchers to segment surfaces into patches which are convex, concave, hyperbolic, parabolic, or planar [7,13].

One of the basic problems to be solved is how to combine profiles from multiple views. In general, there is no way to identify a point on one profile with corresponding points on a profile from a different view, since for smooth surfaces they will not have any points in common. In fact, most stereo algorithms which are based on correspondence find the most similar point and assume it is the same. However, if the camera motion is known, then there is a method to identify points on two different profiles. In our work, the camera has been restricted to planar motion, so that planes parallel to the plane of motion induce a correspondence between the profiles. Nevertheless, it is possible for the profile to change qualitatively from one view to the next, and in order to understand this, the analogous problem for a curve in the plane has been analyzed. These view transitions create ambiguities in the reconstruction process. The criterion used to resolve this ambiguity is that the most likely solution is the one which minimizes the change in depth between adjacent views.

The mathematical approach to this problem is that a smooth surface without inflection points is the envelope of all of its tangent planes. However, there are two problems with this: how to compute the envelope of a family of planes and how to handle inflection points. With the assumption of planar camera motion, the envelope of planes problem has been reduced to that of computing the envelope of a family of lines in a plane, which Giblin and Weiss were able to solve. The algorithm has been applied experimentally to synthetic, noise-free data to reconstruct curves from their profiles with a high degree of accuracy. Future experiments for computing both a dense depth map and Gauss and mean curvatures will employ real data .

4 Mobile Robot Navigation

Vision-based mobile robot navigation is a relatively recent addition to the VISIONS research group at UMass. We have acquired a mobile robot that will enable us to develop a testbed for many of the vision algorithms we have and continue to develop. The robot is to be operated both indoors and out, providing a wide variety of scenes for analysis.

The UMass Autonomous Robot Architecture (AuRA) is being developed to support this research effort. It incorporates both global and reflexive schema-based path planning strategies and utilizes a priori knowledge stored in long-term memory, when available, to assist the vehicle's attainment of its navigational goals.

The chief navigational issues addressed include path following, landmark recognition for vehicle localization and obstacle avoidance. A new fast line finding algorithm is being used for hall and sidewalk navigation and will also be used for localization purposes. A depth-from-motion algorithm developed by Bharwani, Hanson and Riseman [8] is nearly completed and will be used initially for obstacle avoidance. It can also provide information for landmark identification when coupled with top-down knowledge of expected landmark locations. A new fast region segmentation algorithm has found potential application in both path following and vehicle localization. A description of all these algorithms and their use within AuRA can be found in [4] included in these proceedings.

Path planning is handled at two levels. First, the computation of a global path is conducted based on information stored in long-term memory in the form of a meadow-map. An A^* search algorithm capable of dealing with the multiple terrain types found in the map is used to determine the initial route. Then information contained within the map is used to provide appropriate motor behaviors (motor schemas) to enable the robot to attain its navigational goals. Multiple concurrent processes, developed only in simulation thus far, provide the velocity vectors that constrain the robot's motion. Motor schemas afford a relatively straightforward mechanism, using a potential field methodology, for the combination of the outputs of individual motor tasks. These can readily reflect the uncertainty of the perceived environmental objects.

Our Gould system's pipeline parallel processing capabilities is currently being used for rapid application of look-up tables in both the line finding and region extraction algorithms. The acquisition of parallel hardware, a sequent multiprocessor, will decrease the processing time required for both vision and motor tasks is expected to enhance the real-time capabilities of the mobile robot project. When the UMass Image Understanding Architecture [43] is complete, much of the VISIONS system can be migrated directly into AuRA for real-time visual perception.

The successes in actual robot experimentation to date are modest. Successful navigation of both an outdoor sidewalk and an indoor hall using the line-finding algorithm has been achieved. The algorithm is quite robust working with (unchanging) environments in the presence of significant path edge discontinuities (doorways, vehicle tracks, clutter etc.). Obstacle avoidance on vehicle runs has been handled using ultrasonic data thus far. Dead-reckoning information is used minimally in our system as our goal is to serve as a testbed for vision algorithms.

Short-term goals include the finalization of the depth-from-motion algorithm in a form that is useful for obstacle avoidance applications. This algorithm is in the process of being transferred to the Carnegie-Mellon University vehicle navigation prospect (see Motion analysis section of this paper). Our vehicle should be able to navigate cluttered hallways and sidewalks solely using visual data. Installation of a recently acquired UHF transmitter link should be completed soon, allowing the vehicle a greater range than it currently has in its tethered form.

A hierarchical planning system consisting of a mission planner, navigator and pilot is being constructed to handle the task of path planning in both indoor and outdoor environments. Terrain features are taken into account in the determination of the best path for the mobile vehicle. The representations used will include a partial internal model of the environment. This enables the navigator to take advantage of a priori knowledge of the world while the pilot handles unanticipated and unmodeled obstacles as required.

Different path optimization strategies can be used based upon the mission's needs. Whether the safest path, shortest path, or some other metric constitutes the best path will depend on several factors. These would include the nature of the mission, the terrain to be traversed, temporal constraints, energy levels, positional uncertainty, etc. By modeling the free space of the vehicle's world expressly and tying relevant symbolic information to these "meadows", multiple factors are available for path-planning heuristics.

Possibly conflicting sensory input will have to be reconciled using "short-term memory" representations. The meadow map used for navigation will provide regions for instantiation based upon the robot's current position. Information from vision, ultrasonic sensors and positional sensors will be stored in this representation with associated certainty factors that will be altered based upon concurring or contradictory sensor input. This architecture will be sufficiently open-ended to allow the integration of additional sensor modalities (e.g. laser rangefinder, inertial guidance) as they become available.

Spatial and rotational uncertainty regarding the vehicle's position and orientation will be expressly modeled. The resulting spatial error map will be used to guide visual interpretation, windowing the image to reduce the time required for sensory processing. The sensory interpretations then will be used to reshape and reduce the spatial uncertainty map. The feedback provided by the sensors thus restricts the possible positions and orientations of the vehicle, while the probable location of the vehicle is used to guide sensory processing.

Homeostatic control (maintenance of the robot's own internal environment) is another research area. When mobile vehicles become capable of entering hazardous environments and covering longer distances without human monitoring, the status of the robot's energy levels, temperature, and other relevant variables can and should significantly affect planning and action. Through the use of internal sensors (in contrast to environmental sensors), surveillance of the internal state of the robot can be maintained. The information can then be used as necessary to change parameters for motor power consumption, heat production, etc., as well as provide data to the planner for decision making. Any vehicle purported to be "autonomous" must address this issue.

Many of the issues involved in the mobile vehicle research can be seen as complementary to those of other areas in our vision and robotics groups. The use of perceptual and motor schemas in the proposed vehicle architecture exploits many of the concepts used in both the VISIONS scene interpretation group and the work being done for the Laboratory for Perceptual Robotic's distributed programming environment. Multi-sensor integration, certainly crucial for the vehicle's domain, will only benefit from the work being done on the integration of vision, touch and force sensing.

5 Image Understanding Architecture (IUA)

UMass is designing and constructing a highly parallel architecture for computer vision with the goal of achieving real-time processing rates for a knowledge-based system approach to low, intermediate and high level image interpretation tasks. The project involves a joint design and implementation effort with the Hughes Research Laboratory. Our Image Understanding Architecture consists of three tightly coupled layers that correspond to three levels of abstraction. These layers are the Content Addressable Array Parallel Processor (CAAPP) for low-level processing which is a mesh connected array, the Intermediate and Communication as Associate Processor (ICAP) for intermediate vision, and the Symbolic Processing Array (SPA) for high-level processing. Attached to the SPA is a host processor.

As we have previously argued [34,43], an effective computational environment for image understanding requires tight coupling between the portions of the processor responsible for low-, intermediate-, and high-level processing [23,34,43]. In the IUA, the requirements of high-speed, fine-grained bi-directional communication and control is achieved using associative processing techniques to implement three very general processing/communication capabilities:

1. Global Broadcast/Local Compare
2. Some/None Response
3. Count Responders

The CAAPP is 512 x 512 square grid array of 1-bit serial processors intended to perform low-level image processing tasks. The intermediate level is implemented by the Intermediate and Communications Associative Processor (ICAP). The ICAP is also a square grid associative array, of more powerful processing elements; the ICAP is a 64 by 64 array of 16-bit processors. Each ICAP cell is associated with an 8 by 8 tile of CAAPP cells, to which it has access. The SPA processors are powerful, general purpose microprocessors intended for performing high-level symbolic operations, and for controlling sub-array processing in the ICAP and CAAPP arrays. To the SPA, the lower levels appear as an intelligent database that is part of a shared global memory.

5.1 Associative Processing

Associative processing is a technique whereby the processors of the array have the ability to compare sets of data broadcast from a central controller to their own local data. They can then conditionally process both local data and broadcast data based on the results of those tests. Associative processing can best be understood by example, here a single controller (a teacher) interacting with an associative array (a class of students) [20]. If the teacher needs to know if any student in a class has a copy of a particular book the teacher can simply state, "If you have the book, raise your hand." The students each make a check, in parallel, and respond appropriately. This corresponds to a broadcast operation of a controller and a local comparison operation at each pixel in an array, to check for a particular value. Both operations assume that the local processors have some "intelligence" to perform the comparison.

Query and response is just the first part of associative processing, representing a content addressable ("If your hand is up, I'm talking to you.") scheme. To perform associative processing, we must be able to conditionally generate tags based on the value of data and use those tags for further processing. As processing continues only sub-sets of the pixels are involved in any particular operation, but those pixels are operated on in parallel.

The ability to associate tags with values is half the battle for high speed control. We also need to get responses back from the array quickly. Forcing the teacher to sequentially ask each student if they have their hand up defeats the process. The teacher can see immediately if any of the students have their hands up, and can quickly count how many do. Similarly, a Some-response/No-response (Some/None) wire running though the pixel array allows the controller to immediately determine properties about the data in the array, and therefore the state of processing in the array *without looking sequentially at the data values themselves.*

Additionally, fast hardware to perform a count of the responders allows the controller to see summary information about the state of the data in the array. We can write programs that can *conditionally* perform operations based on the state of the computation. By using the properties of the radix representation of numeric values in the array we can use the counting hardware to sum the values in the array. The ability to sum values gives us the power to compute statistical measures such as mean and variance.

These examples of students and pixels illustrate the power of associative processing. We use associative processing as our paradigm of communication in the upwards direction and control in the downwards direction between each pair of levels in the hierarchy. We broadcast criteria for selecting pixels, or regions, or symbolic tokens for selective processing. In this way the higher levels of processing control the lower levels. We test and/or count the response that comes after processing data to allow conditional branching for the next step of processing in a given algorithm. Thus the lower levels provide feedback to higher levels.

5.2 Current Status

At present we are building a 1/64th scale demonstration prototype of the IUA. This is scheduled for completion in early 1988 and will include 4096 CAAPP cells, 64 ICAP cells, a single SPA processor and global controller. The entire prototype will plug into a single-user workstation that will serve as a host.

The prototype is being constructed in 2 micron CMOS technology and will physically consist of a 16-by-20 inch motherboard with 83 daughterboards, 80 of which the daughterboards are 4 by 2.5 inches and the remaining three are 20 by 2.5 inches in size. Of the 80 small daughterboards, 16 are for clock distribution and signal buffering; the other 64 contain the ICAP and CAAPP processors and their memories. The three larger daughterboards provide the controller interface, feedback concentration, and ICAP communications network switching. The motherboard also includes a dual-ported frame buffer memory that allows simultaneous image input and output at video frame rate.

Each processor daughterboard will contain a single custom VLSI chip, a TMS320C25, 256K bytes of static RAM, 384K bytes of dual-ported dynamic RAM, and tri-state bus buffers. The single custom chip holds the 64 CAAPP processors with their local memories, the backing store controller, a refresh controller for the dynamic RAM, and arbitration logic for the various devices that must access the bus of the associated ICAP processor. The custom VLSI chip is currently undergoing fabrication through the MOSIS facility. A first run of the complete custom chip is scheduled for Summer of 1987. Total power dissipation for a processor daughterboard is estimated at approximately 5 watts.

Our software simulator is being re-written to run on an Odyssey signal processing co-processor board in a Texas Instruments Explorer. The Odyssey allows a direct emulation of the ICAP processor and greatly improves the execution times of CAAPP simulations over our VAX-based simulator. The Odyssey simulator will also permit us to closely mimic the interactions of the three processing levels down to the signal level. The Odyssey simulator will initially provide the capability of a single IUA daughterboard, and will eventually be extended to simulate one motherboard.

A VAX-based high-level emulator is also planned for development. Whereas the Odyssey simulator is designed to allow an assembly language level of programming, the VAX emulator will be the vehicle of choice for researchers who wish to get an idea of how the user-level IUA environment will behave. The emulator will sacrifice low-level accuracy in favor of greater speed. For example, the emulator will be restricted to 8, 16 and 32 bit arithmetic, thereby avoiding the bit-serial methods that are actually used in the CAAPP but are very slow in simulation.

Beyond simply testing our hardware design, our ultimate goal for the prototype is to provide a powerful interim development environment for image understanding parallel processing research. A simulated parallel processor is simply too slow to permit any significant amount of experimentation. Once our prototype is up and running, we will be able to accomplish more in the first ten seconds of execution time than we have been able to do in our previous five years of simulation.

Because having this much processing power in a box the size of a personal computer is so attractive, we have designed our prototype to be easily reproducible for a reasonable cost. It has also been designed to be easily adapted to different host systems. We thus hope that it will be possible to construct several copies of the small scale system so that it can be available to a number of researchers prior to construction of the full scale machine.

6 Motion Analysis

Our research in motion analysis has continued with a blend of theoretical and experimental investigations. There has been a concentration on the development of techniques that will find practical use in mobile vehicle navigation. In particular, we are in the process of transferring a motion algorithm from UMass to CMU for recovery of depth under known motion; we expect it to be useful for both obstacle avoidance and landmark recognition. Let us now discuss some of these efforts in somewhat more detail.

Our past motion research concentrated in the recovery of sensor motion parameters from analysis of two images obtained via a sensor in motion. This work was reported in the Ph.D. dissertation research of Lawton [31,32] and Adiv [1]. More recently Pavlin [38] has evaluated the Lawton algorithm for translational motion and determined that the algorithm can be applied effectively with analysis of only 8 to 16 image points between frames if the sensor is pointed approximately in the direction of sensor motion. In addition he has speeded up the algorithm and made it more robust by improving the FOE search algorithm. This was accomplished by computing the error measure for the assumed FOE from a sparser sampling of the visual field (or a more restricted area if constraints on the possible location of the FOE is available). Then, a smooth surface is fit to the error values at those points and the computed minimum of this surface is used to focus the search in the next step of an iterative search process.

Bharwani et al [8,9,10] has continued to develop an algorithm that will compute increasingly more accurate depth information from a sequence of frames derived via approximately known translational motion of the sensor. This algorithm is intended to be applied after FOE recovery using the Lawton-Pavlin algorithm, or when vehicle instrumentation supplies sensor motion. The algorithm matches points between frames up to some match resolution, computes a depth range for the environmental point, and then uses this information to predict a smaller search window in future frames, which then can be searched with finer match resolution and consequently more accurate depth. An important characteristic of this algorithm is that the temporal depth refinement can be applied at a constant computational rate and therefore is well-suited for robot navigation. Since the last report on this algorithm, [10] it has been modified to include the implications of Snyder's theoretical treatment of uncertainty [41] discussed below. Because the FOE and an image point/feature in the first frame actually have an uncertainty region that is two dimensional (at a minimum due to digitization error), the search region must also be two-dimensional. This modification has improved the robustness of the algorithm. In addition the shape of the error surface [3] can be used to dynamically control the resolution of the depth refinement process to experimentally measurable limits.

The two algorithms, FOE recovery and temporal depth refinement, are being packaged into a motion analysis subsystem for use in both the UMass and CMU mobile vehicle efforts. The goal is the analysis of an ongoing sequence of frames from a vehicle in motion to determine obstacles in the path of motion. At CMU it is hoped that this subsystem will operate effectively at a range beyond the useful range of the ERIM sensor (40 foot limit). There are three very general stages of processing that will be briefly discussed. First, frames must be registered since the camera will not be independently stabilized and therefore jerks, bumps, rocking, etc. will introduce local random translational and rotational motion between frames even when the vehicle is undergoing approximate pure global translation. Registration is currently our major problem, and thus for only having a simple registration scheme involving the selection of distinctive points (high contrast and high curvature) that are at a great distance (near horizon) and thus will allow subtraction of the rotational component. Then the FOE will be recovered via the Lawton-Pavlin algorithm using a small number of distinctive points, say 8, in the foreground (10-40 feet). Then the depth of distinctive points in the path of the vehicle will be computed. Finally, either point sets that imply vertical surfaces, or individual points that are not consistent with lying on a planar road surface will be flagged for higher level navigational attention.

Snyder [41,42], has theoretically examined the problem of uncertainty of image measurements in correspondence-based techniques, and their impact in stereo and motion analysis. The location of image features or points are often determined only approximately due to the effect of processing with a window (e.g. as in computing interest operators or using convolution windows) or the result of more complicated processes as in FOE recovery. At a minimum there is sub-pixel uncertainty ($\pm 1/2$ pixel) due to digitization. Uncertainty in such image locations leads to uncertainty in the recovery of depth from both stereo and motion, defines limits to the effectiveness of recovering depth of environmental points or detecting the presence of independently moving objects, and provides the means to determine the relative efficacy between stereo and motion analysis in varying situations. The analysis provides strategies for intelligently controlling the application of stereo and motion algorithms and determining uncertainty ranges for the results that are extracted.

Glazer's recently completed thesis [21] presents an approach to motion detection using multi-resolution methods in a hierarchical processing architecture. Two motion detection algorithms are developed and analyzed. The hierarchical correlation algorithm utilizes a coarse-to-fine control strategy across the resolution levels and overcomes two disadvantages of single-level correlation: large search areas requiring expensive searches and repetitive image structures which cause incorrect matches. The hierarchical gradient-based algorithm [22], generated over low-pass image pyramids, extends single-level gradient algorithms to the computation of large displacements. Within each level the next refinement of the displacement field is obtained by combining a local intensity constraint and a global smoothness constraints. The mathematical formulation involves the minimization of an error functional consisting of two terms, corresponding to the intensity and the smoothness constraints mentioned above. The minimization problem is solved using the finite-difference approach which leads to a multi-resolution relaxation algorithm. A formal analysis of the hierarchical gradient algorithm is presented, including the basic equations for computing a refined disparity vector, the discrete representations and computations for solving these equations, and a geometric interpretation of the resulting relaxation algorithm. The experimental results show that the two algorithms have comparable accuracy and a cost analysis shows that the hierarchical gradient algorithm is less costly.

In his recently completed doctoral dissertation [2] Anandan provides a unified framework for extracting a dense displacement field from a pair of images, as well as an integrated system which is based on a matching approach. This framework appears to be sufficiently general to encompass both gradient-based and correlation matching approaches. It consists of a hierarchical scale-based matching scheme using bandpass filters, orientation-dependent confidence measures, and a smoothness constraint for propagating reliable displacements. His integrated system for the extraction of displacement fields uses the minimization of the sum-of-squarred-differences (SSD) as the local match-criterion, computes confidence measures based on the shape of the SSD surface, and formulates the smoothness assumption as the minimization of an error functional, and overcomes many of the difficult problems that exist in other techniques. The error functional consists of two terms: one of which is called the approximation error, measuring how well a given displacement field approximates the local match estimates, while the other is called the smoothness error, measuring the global spatial-variation of a given displacement field. The finite-element method is used to solve the minimization problem. The approach also gives information for extracting occlusion boundaries in some situations.

Anandan has also shown that the functional minimization problem formulated in his matching technique converges to the minimization problem used in gradient-based techniques (e.g. Glazer's technique mentioned above). In particular, by relating an approximation error functional used in his matching approach to the intensity constraints used in the gradient-based approaches, he explicitly identifies confidence measures which have thus far been implicitly used in the gradient-based approach. Finally, he suggests the ways that algorithms operating on a pair of frames can be developed into multiple-frame algorithms, while discussing their relationship to spatio-temporal energy models.

7 Low-Level Vision

While low-level vision is not the main focus of our research program, almost any large group working on intermediate and high-level computer vision will be engaged in some aspects of low-level vision to support the other efforts. There are several basic segmentation algorithms that our knowledge-based vision research relies upon: histogram-based region segmentation [30,35], straight line extraction [14], and more recently an algorithm for grouping nearby co-linear edges [11]. Analysis of the output of these algorithms has led to several additional investigations. Interesting work is being directed towards edge and line algorithms, as well as texture extraction.

The output of both of our straight line algorithms has made it very obvious that short lines are a very effective mechanism for extracting textured areas and texture descriptions. Since each line has a set of attributes including orientation, length, contrast, etc. they can be filtered or grouped in terms of a variety of features. This may lead to interesting ways to directly extract and characterize textured areas. Alternatively, lines may be used to provide regions with texture characteristics.

An algorithm for grouping edges into curved line segments has continued to be developed and has yielded some promising results [18]. It may be integrated with the straight line algorithm by choosing the output from each representation that is most appropriate. Thus, parameterized curves might replace a piecewise linear sets of edges in the intermediate representation that is examined by knowledge-based interpretation processes.

The algorithm for extracting straight lines by grouping on gradient orientation [14] is being expanded by Reynolds to work on color information rather than intensity. Thus, areas of an image with similar intensity but different color might not be detected by the original algorithm. However, by computing orientation in 3-dimensional color space, edges can be labelled with both their orientation and the colors on either side of the edge. In fact this leads to a straight line extraction algorithm where the line segments represent edges which delimit the boundary between regions of approximately constant color; i.e. instead of a line segment being defined by a gradient magnitude threshold or uniform gradient orientation, the constancy of color contrast across the boundary can also be employed. An additional effort to group similar color edges into textured areas is also being investigated.

Finally, Kitchen and Malin [27] have completed a study of the effect of spatial discretization on the magnitude and direction response of various simple edge operators. They investigate the errors as the true subpixel location of an ideal step edge is varied. Their results show a potentially significant variation can occur in edge magnitude and orientation. They include suggestions for possible improvements of edge operators based upon their techniques.

8 References

REFERENCES

[1] G. Adiv, "Interpreting Optical Flow", Ph.D. Dissertation, Computer and Information Science Department, University of Massachusetts at Amherst, September 1985. Also COINS Technical Report 85-35.

[2] P. Anandan, "Measuring Visual Motion From Image Sequences", Ph.D. Dissertation, University of Massachusetts at Amherst, 1987.

[3] P. Anandan, "A Unified Perspective on Computational Methods for the Measurement of Visual Motion", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[4] R. C. Arkin, E. M. Riseman, and A. R. Hanson, "AuRA: An Architecture for Vision-Based Robot Navigation", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[5] R. Belknap, E. Riseman, and A. Hanson, "The Information Fusion Problem and Rule-Based Hypotheses Applied To Complex Aggregations of Image Events", COINS Technical Report, University of Massachusetts at Amherst, in preparation, 1987.

[6] R. Belknap, E. Riseman, and A. Hanson, "The Information Fusion Problem and Rule-Based Hypotheses Applied to Complex Aggregations of Image Events", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Miami, FL, June 22-26, 1986, pp. 227-234.

[7] P.J. Besl and R.C. Jain, "Segmentation Through Symbolic Surface Description", *Proceedings of CVPR86*, Miami FL, June 1986, pp. 77-85.

[8] S. Bharwani, E. Riseman, and A. Hanson, "Multiframe Computation of Accurate Depth Maps Using Uncertainty Analysis", forthcoming technical report, Computer and Information Science Department, University of Massachusetts at Amherst.

[9] S. Bharwani, E. Riseman, and A. Hanson, "Refinement of Environmental Depth Maps over Multiple Frames", *Proceedings of the IEEE Workshop on Motion: Representation and Analysis*, Charleston, SC, May 7-9, 1986, pp. 73-80.

[10] S. Bharwani, E. Riseman, and A. Hanson, "Refinement of Environmental Depth Maps over Multiple Frames", *Proc. of the DARPA Image Understanding Workshop*, Miami Beach, FL, December 1985.

[11] R. Weiss and M. Boldt, "Geometric Grouping Applied to Straight Lines", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Miami, FL, June 22-26, 1986, pp. 489-495.

[12] M. Boldt and R. Weiss, "Geometric Grouping Applied to Straight Lines", forthcoming technical report, Computer and Information Science Department, University of Massachusetts at Amherst.

[13] M. Brady, J. Ponce, A. Yuille, and H. Asada, "Describing Surfaces", *Proceedings of the 2nd International Symposium on Robotics Research*, Hanafusa and Inoue (Eds.), MIT Press, Cambridge, MA

[14] J. B. Burns, A. R. Hanson, and E. M. Riseman, "Extracting Straight Lines", *IEEE Transactions on Pattern Analysis and Machine Intelligence 8*, No. 4, July 1986, pp. 425-455. Also COINS Technical Report 84-29, University of Massachusetts at Amherst, December 1984.

[15] J. Callahan and R. Weiss, "A Model for Describing Surface Shape", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, San Francisco, June 1985, pp. 240-245.

[16] J. Callahan, "Local Views of a Piecewise Smooth Surface", in preparation.

[17] A. P. Dempster, "A Generalization of Bayesian Inference", *Journal of the Royal Statistical Society*, Series B, Vol. 30, 1968, pp. 205-247.

[18] J. Dolan, G. Reynolds, and L. Kitchen, "Piecewise Circular Description of Image Curves Using Constancy of Grey-level Curvature", COINS Technical Report 86-33, University of Massachusetts at Amherst, July 1986.

[19] B. Draper, R. Collins, J. Brolio, J. Griffith, A. Hanson, and E. Riseman, "Tools and Experiments in the Knowledge-Based Interpretation of Road Scenes", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[20] C. C. Foster, "Content Addressable Parallel Processors", Van Nostrand Reinhold, New York, 1976.

[21] F. Glazer, "Hierarchical Motion Detection", Ph.D. Dissertation, Computer and Information Science Department, University of Massachusetts at Amherst, 1987.

[22] F. Glazer, "Hierarchical Gradient-Based Motion Detection", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[23] A. Hanson and E. Riseman, "The VISIONS Image Understanding System - 1986", in *Advances in Computer Vision*, (Chris Brown, Ed.), Erlbaum Press, 1987.

[24] A. Hanson and E. Riseman, "A Methodology for the Development of General Knowledge-Based Vision Systems", to appear in *Vision, Brain, and Cooperative Computation,* (M. Arbib and A. Hanson, Eds.), 1987, MIT Press Cambridge, MA. Also COINS Technical Report 86-27, University of Massachusetts at Amherst, July 1986.

[25] K. Kanatani, "The Constraints on Images of Rectangular Polyhedra", *PAMI*, 8, No. 4, pp. 456-463, 1986.

[26] Y. L. Kergosien, "La famille des projections orthogonales d'une surface et ses singularities", *C.R. Acad. Sc. Paris*, pp. 929-932, 1981.

[27] L. Kitchen and J. Malin, "The effect of spatial discretization on the magnitude and direction response of simple differential edge operations on a step edge. Part 1: square pixel receptive fields", forthcoming technical report, Computer and Information Science Department, University of Massachusetts at Amherst.

[28] J. J. Koenderink, "What Does the Occluding Contour Tell us About Solid Shape?", *Perception*, vol 13, 1984, pp. 321-330.

[29] C. Kohl, A. Hanson, and E. Riseman, "A Goal-Directed Intermediate Level Executive for Image Interpretation", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[30] R. R. Kohler, "A Segmentation System Based on Thresholding", *Computer Graphics and Image Processing*, **15**, 1981, pp. 319-338.

[31] D. T. Lawton, "Processing Dynamic Image Sequences from a Moving Sensor", Ph.D. Dissertation, Computer and Information Science Department, University of Massachusetts at Amherst, 1984. Also COINS Technical Report 84-05.

[32] D. T. Lawton, "Processing Translational Motion Sequences", *Computer Graphics and Image Processing*, Vol 22, pp. 116-144, 1983.

[33] N. Lehrer, G. Reynolds, and J. Griffith, "Initial Hypothesis Formation in Image Understanding Using an Automatically Generated Knowledge Base", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[34] S. Levitan, C. Weems, A. Hanson, and E. Riseman, "The UMass Image Understanding Architecture", in *Pyramid Multi-Computers*, (Leonard Uhr, Ed.), Academic Press, New York, 1987.

[35] P. A. Nagin, "Studies in Image Segmentation Algorithms Based on Histogram Clustering and Relaxation", COINS Technical Report 79-15, University of Massachusetts at Amherst, September 1979.

[36] H. Nakatani, R. Weiss, and E. Riseman, "Application of Vanishing Points to 3D Measurement", *SPIE International Symposium on Optics and Electro-Optics*, 1984.

[37] H. Nakatani, "Reconstruction of Three-Dimensional Shape Using Pictorial Depth Cues", Ph.D. Thesis, Osaka University, 1986.

[38] I. Pavlin, E. Riseman, and A. Hanson, "A Translational Motion Algorithm Using Hierarchical Search with Smoothing", forthcoming technical report, Computer and Information Science Department, University of Massachusetts at Amherst.

[39] G. Reynolds and R. Beveridge, "Searching for Geometric Structure in Images of Natural Scenes", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[40] G. Shafer, "A Mathematical Theory of Evidence", Princeton University Press, 1976.

[41] M. Snyder, "Uncertainty Analysis in Image Measurements", forthcoming technical report, Computer and Information Science Department, University of Massachusetts at Amherst.

[42] M. Snyder, "Uncertainty Analysis in Image Measurements", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[43] C. Weems, S. Levitan, E. Riseman, and A. Hanson, "The Image Understanding Architecture", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[44] R. Weiss and P. Giblin, "On the Reconstruction of Surfaces From Their Profiles", *Proceedings of the DARPA Image Understanding Workshop*, Los Angeles, CA, January 1987.

[45] T.E. Weymouth, "Using Object Descriptions in a Schema Network For Machine Vision", Ph.D. Dissertation, Computer and Information Science Department, University of Massachusetts at Amherst. Also COINS Technical Report 86-24, University of Massachusetts at Amherst, 1986.