Progress in Computer Vision at the University of Massachusetts

Allen R. Hanson
Edward M. Riseman
Charles C. Weems

CMPSCI TR93-62

July 1993

This work is supported in part by the Advanced Research Projects Agency (via TACOM), HDL contract number DAAL02-91-K-0047, and TEC contract number DACA76-92-C-0041, by the National Science Foundation under grant numbers CDA-8922572, IRI-9113690 and IRI-9208920, and by RADC under contract number F30602-91-C-0037.

Progress in Computer Vision at the University of Massachusetts

Allen R. Hanson, Edward M. Riseman, and Charles A. Weems
Computer Vision Research Laboratory
Dept. of Computer and Information Science
University of Massachusetts
Amherst, MA 01003

Abstract[1]

This report summarizes progress in image understanding research at the University of Massachusetts over the past year. Many of the individual efforts discussed in this paper are further developed in other papers in this proceedings. The summary is organized into several areas:

1. Mobile Robot Navigation
2. Motion Analysis
3. Interpretation of Static Scenes
4. Image Understanding Architecture
5. RADIUS Image Exploitation

The research program in computer vision at UMass has as one of its goals the integration of a diverse set of research efforts into a system that is ultimately intended to achieve real-time image interpretation in a variety of vision applications.

1. Mobile Robot Navigation

1.1. Automated Model Acquisition and Extension

The focus of the UMass mobile robot navigation project is robust landmark-based navigation, with a focus on automated model acquisition and model extension. Thus, for navigation in unmodelled or sparsely modelled environments, our general scenario would involve the initial acquisition of prominent visual features that can serve as landmarks. This initial phase of partial model acquisition is necessary because there are few situations where a model of a complex outdoor scene will be available a priori. Once a sparse model is available, then the vehicle position and orientation (i.e. pose) can be recovered by recognizing landmarks. The model extension phase involves tracking new unmodelled features (points and/or lines), and using the landmarks and partial model to determine the camera pose for triangulation of the new features and incorporation into the 3D model.

Most of the algorithms have been described in previous IUW proceedings and the general vision literature [Beveridge 92, Kumar 92, Sawhney 92, 93]. These algorithms have been shown to be very accurate in many indoor experiments using a camera mounted on a mobile robot and on a moving robot arm. One new experiment that integrated several components involved the detection of shallow structures - an aggregatation of line features that can be approximated in an image sequence as a frontal planar surface. The 3D position of these features served as the acquired model, with a depth error of less than 4%. As motion of the camera continues, the model is extended with depth information on other tracked points to accuracies of less than 2% error in depth.

[1]This research has been supported in part by the Defense Advanced Research Projects Agency under TACOM contract number DAAE07-91-C-R035, HDL contract number DAAL02-91-K-0047, and TEC contract number DACA76-92-C-0041, by the National Science Foundation under grant CDA-8922572, IRI-9113690, and IRI-9208920, and by RADC under contract number F30602-91-C-0037.

1.2. Status of the UMass Mobile Perception Laboratory (MPL)

1.2.1. Physical Description

The UMass Mobile Perception Laboratory (MPL) is based on a significantly modified HMMWV. The design of the overall system includes actuators and encoders for the throttle, steering column and brakes that closely match those being used by CMU, controlled by 68020's in a 6u VME cage. The low-level control software for controlling speed and steering angle will also be the same as that of CMU. The modifications and component installation is being performed by RedZone, Inc., a Pittsburgh-based firm specializing in custom robotics, and was completed at the beginning of February 1993.

Electrical power is supplied by a 10kW diesel generator, whose output is split into two 5kW circuits. The first circuit is conditioned and backed by a 5kW uninterruptible power supply (UPS) system, and is used to supply power to all sensitive electronic equipment. The second circuit is not conditioned and is used to power the air conditioners. Both circuits are attached to a shore-power hook-up that provide an alternative power source to the on-board generator.

The physical lay-out of equipment was designed to

1) provide for two on-board programmer stations,

2) minimize destructive modifications to the body of the vehicle, and

3) keep the center of gravity as far forward as as possible, in order to minimize stress on the suspension system.

The first programmer station is located in the HMMWV's passenger seat, with a 17" color x-terminal fixed to the metal platform between the passenger's and driver's seats. The second programmer station is located behind and slightly above the driver, and includes a car seat, mounting brackets for both an SGI color terminal and a small SONY monitor for viewing raw TV signals.

The back of the vehicle is filled with equipment. On the driver's side of the vehicle, behind the second programmer station, is all equipment associated with providing power. On the passenger's side there are four enclosed, air conditioned 19" computer frames for the on-board computer systems. The first frame will hold the 6u VME cage for throttle, brake and steering controllers and a second 6u VME cage for holding digitizers, image frame stores and a Datacube MaxVideo20. The second computing frame will contain a 9u cage for the Silicon Graphics four-node multiprocessor, as well as the SGI's disk drives, power supply and (removable) tape drive. The third frame is reserved for the Image Understanding Architecture (IUA). The fourth frame is for future additions, including video recorders for collecting data and recording experiments. Together, the four frames take up the length of the vehicle's bed, as do the programmer station, UPS cage and generator on the left side.

1.2.2. Sensor Configuration

The vehicle's sensor package includes a Staget, which is a rotating stabilized platform being supplied to the UMass and CMU vehicles by TACOM. The UMass Staget is mounted on a level platform located at the center of the roof of the cab. We are planning to put two CCD color cameras on the Staget, one with a wide angle lens and the other with a telephoto lens. The first will be used to locate landmarks in the larger scene, and the second will be used for landmark matching and accurate pose refinement. The Staget will also contain a FLIR sensor. The Staget's hardware is mounted above the driver's head in the enclosure originally occupied by the HMMWV's NBC system. Forward of the Staget, at the edge of the cab's roof, is a long (5" by 12" by 12") rectangular enclosure with a glass front and hinged roof for forward-looking stereo cameras.

1.2.3. Software Environment

MPL is an experimental laboratory for testing and integrating different approaches to problems in autonomous navigation, including, but not limited to, landmark-based navigation, obstacle detection and avoidance, model acquisition, and road following. It is therefore important that MPL have a software environment where multiple visual modules, addressing different subtasks, can be easily integrated, and where researchers can quickly experiment with different combinations and parameterizations of those modules. At the same time, MPL's software environment must be efficient enough to meet the demands of real-time navigation research.

The need to balance between flexibility and efficiency has led us to design a software environment with two major components: the ISR3 in-memory data store, and a graphical programming interface adapted from Khoros. ISR3 is the glue that binds independent visual modules together [Draper 93a]. It is an in-memory database that allows users to define structures for storing visual data, such as images, lines and surfaces. ISR3 then serves as a buffer, so that, for example, lines produced by one module can be used by another, even if the second module is run later or on a different processor than the first. ISR3 also provides modules with efficient spatial access routines for visual data, and protects data from being simultaneously modified by two or more concurrent processes. The graphical programming interface allows programmers to easily sequence modules and modify their parameters.

1.2.4. Navigation System

A preliminary version of a behaviour-based system for determining vehicle pose from known landmarks has been designed. It is assumed that pose estimates and associated covariance (error) estimates are returned from several subsystems (GPS, INS, Landmarks, and dead reckoning) asynchronously. These estimates are continually combined via a Kalman filter into a single pose estimate (and associated covariance matrix estimate) and stored in a vehicle state vector. The vehicle pose error is continually monitored in a simple loop which branches to a behavior selection strategy when the vehicle pose error exceeds a preset threshold.

The system also contains a video image frame buffer and STAGET control subsystem. This system maintains image and pose temporal histories (time-stamped images and corresponding pose estimates) in a fixed-length first-in last-out queue. This information is available to the remainder of the system. The STAGET control interface permits the STAGET to be repositioned relative to the vehicle and maintains information about the various STAGET parameters and conditions, including information about the current lens aperture and focal length.

All the landmark matching and pose refinement algorithms have been tested extensively, although to a great extent only in indoor domains. A large portion of the original LISP has been ported to C. The plan for the coming year of research is to develop the following behaviors: road following, obstacle avoidance, landmark detection, landmark tracking, and model extension.

Initially, two types of landmark processing behavior will be specified. The first behavior for landmark tracking assumes that a landmark (or set of landmarks) are currently being tracked via the STAGET and all that is necessary is that the vehicle pose be recomputed from the tracked landmarks. However, there are computational tradeoffs as a function of the speed of the vehicle, and the distance and number of landmarks. Thus, not all landmarks may be tracked frame by frame.

The second landmark navigation behavior assumes that no landmarks are currently being tracked and therefore a new landmark must be acquired. This will involve access to a stored 3D model of the campus environment (which initially has been constructed a priori) in order to control the Staget and window on subimages via the Staget. However, the availability and density of landmarks will vary significantly in different areas of the test environment, and therefore model extension will be a necessary goal. Ultimately we seek to demonstrate that an accurate 3D model of the environment can be acquired via exploration in a purely bottom-up manner, while carrying out independent goal-oriented navigation tasks.

1.3. Qualitative Navigation via Image-Based Homing

If the world changes or the robot fails to recognize a landmark, the robot's perception of the world will not correspond to its current map of the world. However, there is ambiguity in whether the errors are in its perception or its map, and if the latter, it must update its map.

Pinette [Pinette 91] has been developing a principled approach to automatic map construction and maintenance. In place of the usual construction of a geometric map, snapshots of the world at selected target locations along the route are stored as the robot's knowledge of that path. By noting places where a set of memorized routes intersect, a topological "road map"of routes and junctions are represented. To retrace a stored route, a qualitative homing algorithm based on purely local visual servoing is employed to home between successive target locations along the route. This homing algorithm uses no geometric model or positional information; rather, it servos directly on the stored image for a target location, choosing headings that reduce the difference between features of the current bearings and those in the target snapshot. A "consistency-filtering" algorithm has been developed for handling incorrectly matched landmark features [Pinette 92]. It is shown that this algorithm guarantees reliable homing as long as more than two-thirds of the landmarks are correctly identified.

A very robust implementation of a robot navigation system has been developed using image-based homing with a spherical mirror for encoding a 360 degree view at each target location. This navigation system has been implemented as part of an indoor manufacturing automation application domain. It is not yet clear whether these techniques are directly applicable to unconstrained outdoor domains and large-scale space.

2. Motion Analysis

2.1. Multi-Frame Structure from Motion

In robot navigation a model of the environment needs to be reconstructed for various applications, including path planning, obstacle avoidance and determining where the robot is located. Traditionally, the model was acquired using two images (two-frame Structure from Motion) but the acquired models were unreliable and inaccurate. Generally, research has shifted to using several frames (multi-frame Structure from Motion) instead of just two frames. However, almost none of the reported multi-frame algorithms have produced accurate and stable reconstructions for general robot motion. The main reason seems to be that the primary source of error in the reconstruction - the error in the underlying motion - has been mostly ignored. Intuitively, if a reconstruction of the scene is made up of points, this motion error affects each reconstructed point in a systematic way. For example, if the translation of the robot is erroneous in a certain direction, all the reconstructed points would be shifted along the same direction.

Recently, Thomas [Thomas 93a,b] has mathematically isolated the effect of the motion error (as correlations in the structure error) and has shown theoretically that including these correlations in the computation can dramatically improve existing multi-frame Structure from Motion techniques. In several experiments on our indoor robot, the environmental depths of points from 15 to 50 feet away from the camera (and for which ground truth data was available) were reconstructed with errors in the 1-3% range. In one further experiment, the multi-frame full-correlation algorithm was first used to create a model (a set of points) of an indoor hallway from several initial frames of image data. This model was then used to compute the pose of the robot over subsequent frames using Kumar's pose recovery algorithm. The estimated robot pose and actual robot position in the hallway differed by a maximum of three to four inches over a 12.8 foot path.

2.2. Recovering Affine Transforms from Image Sequences

Deformations due to relative motion between an observer and an object may be used to infer 3-D structure. Up to first order these deformations can be written in terms of an affine transform. The recovery of an affine approximation to image deformation has recently been the focus of a large amount of research, and has found application in such disparate areas of computer vision as image stabilization, optical flow computation and segmentation, structure from motion, stereo, and texture, and obstacle avoidance.

Manmatha [Manmatha 93] has developed a technique for measuring the affine transform locally between two image patches using weighted moments of brightness. Unlike previous methods, this technique correctly handles the problem of finding the correspondence between deformed image patches, as is necessary for a correct computation of the affine transform. It is capable of determining affine transforms of arbitrary size, whereas most previous approaches are limited to small transforms. It is first shown that the moments of image patches are related through functions of affine transforms. Finding the weighted moments is equivalent (for the purposes of measuring the affine transform) to filtering the images with gaussians and derivatives of gaussians. In the special case where the affine transform can be written as a scale change and an in-plane rotation, the zeroth and first moment equations are solved for the scale. In experiments on synthetic and real images for this case, the scale was recovered robustly and shown to give reliable depth estimates. Work is continuing on extending the basic techniques to the general case.

2.3. Multi-Sensor Dextrous Manipulation

Grupen and Weiss [Grupen 93] have continued their work on a multi-sensor approach to dextrous manipulation. The goal of this project is the integration of sensing and control for the task of finding a stable grasp configuration for an unknown object. A subgoal is the integration of visual and haptic (proprioceptive) sensory data to incrementally build a model of the object. This approach uses knowledge of the task and the accuracy and completeness of the model to control the sensing actions.

The system consists of a camera mounted on one robot and the Utah/MIT hand mounted on another. The system calibration or identification problem involves computing the transformation from the coordinate system defined by the manipulator robot to the coordinate system defined by the camera robot. The pose determination algorithm of Kumar and Hanson [Kumar 92] has been adapted for this purpose. As the manipulator robot moves, known feature points are tracked. Given the kinematics of this robot, the pose of the camera with respect to the coordinate frame of the manipulator robot are computed and incrementally refined using iterative, extended Kalman filtering. Experiments were performed to demonstrate that the accuracy of the filtering algorithm was comparable to that of smoothing using a least squares fit with all of the data, yet the computation time was much less. An additional feature of the method is that the kinematics of the camera robot can be computed at the same time.

Grupen and Huber [Huber 92] have obtained 3D surface points from the Utah/MIT hand without the use of tactile sensors. The measurements used are posture, velocities, and torques. This will be integrated with the measurements obtained from the camera sensor.

2.4. Shape Recovery from Occluding Contours

Recovering the shape of an object from two views (e.g. stereo) fails at occluding contours of smooth objects because the extremal contours are view dependent. For three or more views, shape recovery is possible, and several algorithms have recently been developed for this purpose. Szeliski and Weiss [Szeliski 93] have developed a new approach to the multiframe shape recovery problem which does not depend on differential measurements in the image, which may be noise sensitive. Instead, a linear smoother is used to optimally combine all of the measurements available at the contours (and other edges) that are tracked through the set of images. This allows the extraction of a robust and dense estimate of surface shape and the integration of shape information from both surface markings and occluding contours. The results provide an extremely promising path for recovery of 3D shape models in an industrial setting where the motion is known.

3. Interpretation of Static Scenes

3.1. Learning 3D Recognition Strategies

Most knowledge-directed vision systems are tailored to recognize a fixed set of objects within a known context. Generally, the programmer or knowledge engineer who constructs them begins with an intuitive notion of how each object might be recognized, a notion which is refined by trial-and-error. Unfortunately, human engineering is not cost-effective for many real-world applications. Moreover, there is no way to ensure the validity of hand-crafted systems. Worst of all, when the domain is changed, the systems often have to be rebuilt from scratch.

The Schema Learning System (SLS) [Draper 92, 93b] automates the construction of knowledge-directed recognition strategies. Starting from a knowledge base of visual procedures and object models, SLS learns robust strategies for locating landmarks in images and recovering their positions and orientations, if necessary. Each strategy is specialized to a landmark, taking advantage of its most distinctive characteristics, whether in terms of color, shape, or contextual relations, to quickly focus its attention on the landmark and recover its pose. Furthermore, because SLS learns from experience by a strict generalization algorithm, it is possible to predict both the expected costs and the expected error rates (due to a lemma by Valiant) of the strategies it develops.

3.2. Figural Completion from Principles of Perceptual Organization

Figural completion is the preattentive ability of the human visual system to build complete and topologically valid representations of environmental surfaces from the fragmentary evidence available in cluttered scenes. A description of a grouping system developed by Williams, employing a two-stage process of completion hypothesis and combinatorial optimization, appeared in a previous workshop proceedings [Williams 90]. Preliminary experimental results were also reported. Since that time there has been significant progress in two major areas. First, the mathematical basis for the grouping constraints employed in the optimization stage has been clearly elucidated. This has allowed a proof of the necessity and sufficiency of the grouping constraints for scenes composed of flat embeddings of orientable surfaces with boundary. Second, a more advanced grouping system which uses cubic Bezier splines of least energy to model the shape of perceptual completions has been implemented. The new system is demonstrated on a number of figures from the visual psychology literature which are beyond the capability of the old system.

3.3. Perceptual Organization of Curvilinear Structure

During the past year, Dolan has continued his work on curvilinear grouping [Dolan 92]. A SIMD implementation of the curvilinear grouping system has been developed, along with a simplified, distributed representation of curves for use in the CAAPP. The integration of multiple grouping processes--in particular, curvilinear and area grouping -- is currently being examined. Many of these ideas are being incorporated in a general grouping module for KBVision, which will facilitate research and experimentation with many diverse forms of grouping.

3.4. Stochastic Projective Geometry

The use of projective invariants for object recognition and scene reconstruction has been the subject of intense interest in the image understanding community over the past few years. Although classic projective geometry was developed with mathematically precise objects in mind, practical applications must deal with errorful measurements extracted from real image sensors. A more robust form of projective geometry is needed, one that allows for possible imprecision in its geometric primitives. In his Ph.D. thesis [Collins 93], Collins represents and manipulates uncertain geometric objects using probability distributions in projective space, allowing valid geometric constructions to be carried out via statistical inference. The result is a methodology for scene reconstruction based on the principles of projective geometry, yet also dealing with uncertainty at a basic level. The effectiveness of this framework has been demonstrated on several geometric problems, including the derivation of 3D line and plane orientations from a single image using vanishing point analysis, the extraction of a planar patch scene model using stereo line correspondences, and the reconstruction of planar surface structure using multiple images taken from unknown viewpoints by uncalibrated cameras.

More specifically, Collins shows that projective N-space can be visualized as the surface of a unit sphere in (N+1)-dimensional Euclidean space. Each point in projective space is represented as a pair of opposing or antipodal points on the sphere. By the identification of projective space with the unit sphere, antipodally symmetric probability distributions on the sphere may be interpreted as probability distributions over the points of projective space, and standard constructions of projective geometry can then be augmented by statistical inferences on the sphere. Probability densities defined in this way can also be used for representing uncertainty in unit vectors, orientations, and the space of 3D rotations (via unit quaternions).

3.5. Shape from Shading

Oliensis' previous work on shape from shading [Oliensis 92] has been extended in a number of ways. First, while our earlier work usually assumed that the illumination was from the direction of the camera, the shape reconstruction algorithms and convergence proofs have been extended more recently to the case of illumination from any direction [Oliensis 93a]. As before, these algorithms are provably and monotonically convergent, and (in many cases) can be shown to converge to the correct surface. Moreover, it has been shown that a whole family of algorithms could be developed, and that all would give equivalent surface reconstructions. This is convenient since some of the algorithms are better for theoretical analysis while others are more efficient in practice. The uniqueness proofs for the surface given the shaded image, and the corollary that regularization is not necessary for shape from shading, have also been extended. Experimentation with these algorithms on synthetic and real images show that they are fast and robust, taking less than 10 seconds on a DECstation 5000 for a 200 x 200 real image.

These algorithms still require that a small amount of information on the surface be provided, namely: 1) a list of those singular points (the brightest image points) corresponding to local minima of the surface height (as opposed to the other possibilities of a local maximum or a saddle point); and 2) the heights of these singular points. However, in a second extension of previous work [Oliensis 93b], Oliensis has developed a new algorithm that is capable of determining this information automatically, and thus can reconstruct a general surface from shading with no a priori information on the surface. In experimental tests on complex synthetic images, this algorithm has produced good surface reconstructions over most of the image. For 128 x128 images, the reconstruction takes less than 30 seconds on a DECstation 5000. Moreover, the algorithm appears noise resistant, giving good reconstructions even in the extreme case of an added pixel noise of 10%. It appears that it will also be possible to prove the convergence of this algorithm to the correct surface in the limit of perfect resolution.

All algorithms thus far have assumed that the imaged surface was matte. Even with this restriction, the algorithms are potentially useful in controlled industrial or research applications. At UMass these algorithms will be ported to the robotics laboratory environment, and used in combination with other means of shape sensing and recovery to aid in research in grasping partially or unmodeled objects. Further extensions include adapting the current algorithms to the realistic case of a partially specular surface. With this extension, shape from shading could become practical for a variety of applications.

4. Image Understanding Architecture (IUA) Overview

Work on the IUA [Weems, 1993] has advanced in three areas in the preceding year: compilers and system software, hardware and architecture, and applications and algorithms. The IUA is a tightly coupled, heterogeneous parallel processor being developed by UMass, Hughes Research Labs, and Amerinex Artificial Intelligence (AAI) under DARPA funding. It is intended to support real-time knowledge-based vision applications and research by providing three distinct parallel processors in a single architecture: a fine-grained SIMD/Multi-associative array for low-level vision, a medium-grained SPMD array for intermediate-level symbolic vision, and a coarse-grained multiprocessor for high-level, knowledge-based processing. A proof of concept prototype of the IUA was constructed under a previous effort and the current work is directed at developing a second generation of the system with enhanced performance and the ability to be fielded in the DARPA Unmanned Ground Vehicles (UGV) program.

4.1. IUA Compilers and System Software

AAI has completed development of the C++ class library for the low level of the IUA. The class library defines a set of image plane types upon which parallel operations may be performed. Work at AAI includes the incorporation of additional optimization code into the Gnu C++ compiler so that image planes are treated more like first-class objects in C++. An automated test system has also been developed for the machine's microcode library, to facilitate regression testing of new releases. For the intermediate-level processor, basic operating system support, multitasking, and messaging have been implemented on a TMS320C30 Single Board Computer (SBC), and recently these were transported to another SBC with two TMS320C40 processors that are configured to simulate the intermediate level of the IUA. A debugger has also been implemented for the intermediate level. Work is now under way to transport the KBVision™ system to the IUA.

UMass has implemented a version of the Apply language for the low-level processor of the second generation IUA. The compiler generates code compatible with the C++ class library. It permits us to easily import image processing operations written for the CMU Warp or Intel iWarp machines.

4.2. IUA Hardware Status

The prototype IUA has been running at Hughes for most of the last year. Under the prototype development contract, only a very simple controller was built to demonstrate the basic functionality of the processor arrays. It was never intended that the prototype controller be fully programmable. However, Hughes and Amerinex AI invested additional effort to develop software that allows C++ code for the second generation to execute on the prototype hardware. Because of the nature of the controller, instructions can only be issued at VME bus rates to the array, which is significantly slower than the array can accept them. However, it does permit demonstration of the functionality of the array hardware on real applications. Hughes has since implemented the low-level portion of the DARPA IU Benchmark, an SDI application, and an ATR application on the prototype.

The custom chips used in the IUA have been fabricated and are undergoing testing. Each chip contains 256 bit-serial processors with on-chip cache, and has roughly 600,000 transistors. The system's array control unit, backplane, and chassis have been built and tested. Processor and memory boards are currently under construction. The I/O subsystem for the machine has also been designed, and will support the equivalent of 20 simultaneous sensor inputs at 512 X 512 X 8-bit resolution with automatic mapping onto the processor virtualization scheme used for the low level, with almost no latency. The I/O subsystem will also support the selection of multiple regions of interest from an image. Hughes has indicated that the first machine should be assembled by the end of February, 1993.

Work has already begun on the analysis and design for the third generation IUA. UMass has developed a system for capturing traces of programs written in the C++ class library as they execute on an abstract parallel machine. The traces are then fed to a simulation system that models hardware architectures with different features and parameters. The system allows us to gather real performance data for different architectural configurations, and to analyze the data statistically. The performance data will then be contrasted with cost estimates for the different configurations to produce a specification for the third generation IUA.

4.3. IUA Applications and Algorithms

The low-level processor of the IUA is a square mesh of processing elements, augmented with a second (reconfigurable) mesh, called the Coterie Network . This network allows the mesh to be partitioned, for example, into areas corresponding to regions in an image. One particularly useful operation is the ability to enumerate elements within a partition or to summarize (reduce) the information in a partition to a single value. The parallel prefix operation is the general form of this type of operation. It is especially desirable to be able to carry out parallel prefix in all partitions at once, i.e. to perform a multi-prefix operation [Herbordt, 1992]. An algorithm has been developed for multi-prefix that is significantly faster than alternatives using general purpose routing in the mesh.

As recommended by the DARPA IU Benchmark Workshop participants, much of the benchmark [Weems, 1988, 1990] has been recoded as a set of library routines which are called by the core of the benchmark. We have also begun developing the second level benchmark, which will incorporate tracking of moving objects over a sequence of images. The goal of the new benchmark is to test system performance over a longer period of time so that, for example, caches and page tables will be filled. The benchmark will also explore I/O and real-time capabilities of the systems under test, and involve more high-level processing.

UMass has developed a parallel algorithm for the IUA that computes a dense depth map for a scene from a pair of images taken by a moving sensor [Dutta 93]. The algorithm has an average error of about 8 percent in depth, as computed from randomly sampling points corresponding to objects in the scene with known distances from 21 to 76 feet from the camera. The experiments were done with fairly large displacements (four feet of forward motion between the images) so that a large (41 X 41 pixel) search window was required to establish correspondences, resulting in 1681 image-to-image correlations being performed. In simulations of the second generation IUA, it was determined that the execution time will be about 0.54 seconds, of which 0.53 seconds is taken up solely by the correlations. We are thus looking into approaches in which an estimate of the motion is available or in which a series of frames with smaller displacements can be used (allowing the search window to be constrained).

4.4. Line Extraction

UMass has also developed a parallel algorithm for extracting straight lines from an image. Using the second generation IUA simulator, the algorithm executes in as little as 31 milliseconds for images that map to the array with a 1:1 virtualization ratio. We are currently evaluating the quality of the results, but a preliminary examination indicates that the algorithm gives very consistent lines over sequences of images, which is an important attribute in the support of algorithms that use line tracking.

5. Image Exploitation under RADIUS

UMass is developing mechanisms for site model acquisition, extension and refinement [Collins 93a] based on technology that has already proven effective in the mobile robotics domain.. Automatically acquiring the initial 3D site models from scratch is a challenging problem that will be the focus of future research. Our current work assumes that a partial model of the site is provided apriori by the image analyst. Our model-based refinement and extension algorithms are then applied to automatically correct inaccuracies in the initial site models, and extend them to include previously unmodeled cultural features (buildings, roads, etc.) based on information from new images.

Rather than building a turn-key system, UMass will be delivering a set of modules for performing specific tasks of direct benefit to the image analyst. The following is a list of the early deliverable modules that are currently being evaluated on the model board test imagery supplied to the research community.

1. Feature Extraction Module. This module condenses the vast amount of information in each image into a manageable set of symbolic descriptions. Two straight line extraction algorithms are being evaluated: the Burns algorithm based on fitting planar patches to the underlying image intensity surface, and the Boldt algorithm for hierarchical geometric edge grouping. Also under development are routines for extracting curved lines, and for locating dihedral and trihedral junctions to subpixel accuracy.

2. Model Matching Module. Given the approximate pose (location and orientation) of the sensor, a partial 3D wireframe site model, and a set of extracted straight lines, the best match of the projected 3D model to the line data will be found using a novel model matching algorithm due to Beveridge.

3. Model Extension Module. Given a partial model and a set of model-data feature correspondences over multiple images, the site model will be extended to include further unmodeled features. Two techniques are being evaluated. The first is based on recovering the camera pose using a robust pose estimation technique due to Kumar. This algorithm is effective even when significant numbers of feature correspondences are in error. Using the computed pose for multiple images from multiple viewpoints, the 3D positions of unmodelled features are found by triangulation. A second approach is based on direct estimation of the 3D to 2D projective transformation relating model features to image features. The benefits of this approach are that multiframe triangulation can still be performed without first solving for camera pose, and without relying on accurate knowledge of the internal camera parameters.

4. Vanishing Point Module. Vanishing point analysis is a flexible tool for geometric reasoning in cultural domains. Among its many uses are the determination of 3D line and plane orientations,

refinement of extracted linear features based on convergence constraints, pose estimation, and camera calibration. An efficient vanishing point detection and estimation algorithm due to Collins and Weiss is being evaluated.

In addition to developing new techniques for automatically acquiring initial site models, new research will investigate statistical techniques for applying projective invariants to the modeling process to accurately derive structure without explicit camera models or knowledge of viewpoint. Initial experiments in this direction have yielded promising results. Other encouraging results have been obtained regarding the difficult problem of image to image registration. A technique based on vanishing point analysis [Collins93b] allows an oblique aerial view to be rectified (unwarped) to present a simulated vertical view, allowing full perspective aerial images to be registered with a computationally tractable affine matching approach.

6. References

Beveridge, J.R., and E. M. Riseman, "Hybrid weak-perspective and full-perspective matching", *Proc. CVPR,* 1992, pp. 432-438.

Collins, R., Allen R. Hanson, Edward M. Riseman, Yong-Qing Cheng, "Model Matching and Extension for Automated 3D Site Modeling", *Proc. 1993 DARPA IUW,* Washington, DC. April 19-21, 1993a., pp. 197-203.

Collins, R., and J. Ross Beveridge, "Matching Perspective Views of Coplanar Structures using Projective Unwarping and Similarity Matching", *Proc. 1993 DARPA IUW,* Washington, DC. April 19-21, 1993b, pp. 459-463.

Collins, R., forthcoming Ph.D. thesis, 1993.

Dolan, J., and E. Riseman, "Computing Curvilinear Structure by Token-Based Grouping", *IEEE CVPR,* Champaign, IL, June 1992, pp. 264-270.

Draper, B., and A. Hanson, "ISR3: A Token Database for Integration of Visual Modules", *Proc. 1993 DARPA IUW,* Washington, DC. April 19-21, 1993a, 1155-1161.

Draper , B., and E. Riseman, "Statistical Properties of Learning Recognition Strategies," *Proc. 1993 DARPA IUW,* Washington, DC. April 19-21, 1993b, pp. 557-565.

Draper, B., A. Hanson, and E. Riseman, "Learning Blackboard-Based Scheduling Algorithms for Computer Vision", Computer Science Department, University of Massachusetts, Technical Report TR92-51, August 1992.

Dutta, R., C. Weems, and E. Riseman, "Parallel Dense Depth from Motion on the Image Understanding Architecture", *Proc. 1993 DARPA IUW* , Washington, DC. April 19-21, 1993, pp. 1145-1153.

Grupen, R., and R. Weiss, "Sensor-Based Incremental Planning for Multifingered Manipulators, Grantees Conference of the NSF Division of Design and Manufacturing", Univ. of North Carolina, January 1993.

Herbordt, M., and C. Weems, "Computing Reduction and Parallel Prefix Using Coterie Structures", *Proc. IEEE Conf. on the Frontiers of Massively Parallel Processing*, October, 1992, pp. 141-149.

Huber, M., and Grupen, R., "2-D Contact Detection and Localization Using Proprioceptive Information", Computer Science Technical Report 92-59, University of Massachusetts, Amherst, August, 1992.

Kumar, R., "Model Dependent Inference of 3D Information from a Sequence of 2D Images", Ph. D. Thesis and Technical Report TR92-04, Department of Computer Science, University of Massachusetts (Amherst), 1992.

Manmatha, R., and J. Oliensis, "Measuring Affine Transforms - I: Recovering Scale and Rotation", *Proc. 1993 DARPA IUW*, Washington, DC. April 19-21, 1993, pp. 449-458.

Oliensis, J., and P. Dupuis, "Provably Convergent Algorithms for Shape from Shading", *Proc. DARPA IUW,* Washington, DC. April 19-21, 1993a, pp. 1121-1130.

Oliensis, J., "A General Algorithm for Shape from Shading", *International Conference on Computer Vision,* Berlin, Germany, May 11-13, 1993b.

Oliensis, J., and P. Dupuis, "Direct Method for Reconstructing Shape from Shading", Computer Science Department, University of Massachusetts, Technical Report TR92-32, May 1992.

Pinette, B., "Qualitative navigation. II: Handling inconsistently-matched landmarks", *Proc. SPIE,* Volume 1831, Mobile Robots VII, Boston, MA. 1992.

Pinette, B., "Qualitative Navigation", *Proceedings of IEEE Symposium on Intelligent Control,* Arlington, VA. 1991, pp. 318-323.

Sawhney, H., "Spatial and Temporal Grouping in the Interpretation of Image Motion", Ph.D. Thesis and Technical Report 92-05, Computer Science Department, University of Massachusetts, Amherst, February, 1992.

Sawhney, H., and A. R. Hanson, "Affine trackability aids obstacle detection", *Proc. CVPR,* Champaign, IL, June 1992, pp. 418-424.

Sawhney, H., R. Kumar, A. Hanson, E. Riseman, "Landmark-Based Navigation - Model Extension and Refinement", *Proc. DARPA IUW,* Washington, DC, April 19-21, 1993, pp. 507-514.

Szeliski , R., and R. Weiss, "Robust Shape Recovery from Occluding Contours Using a Linear Smoother", *Proc. 1993 DARPA IUW* , Washington, DC. April 19-21, 1993, pp. 939-948.

Thomas, J., A. Hanson, and J. Oliensis, "Understanding Noise : The Critical Role of Motion Error in Scene Reconstruction", *Proc. ICCV,* Berlin, Germany, May 11-13, 1993a, (A similar paper appears in these proceedings).

Thomas, J., "Obtaining the Robot Path Using Automatically Acquired Models", *Proc. International Conference on Intelligent Autonomous Systems (IAS),* Pittsburgh, PA, February 1993b, pp. 489-497.

Weems, C., M. Herbordt, R. Dutta, K. Daumueller, G. Weaver, S. Dropsho, J. Burrill, R. Lerner, A. Hough, "Progress and Current Research in the Image Understanding Architecture", *Proc. 1993 DARPA IUW* , Washington, DC. April 19-21, 1993, pp. 1133-1140.

Weems, C.C., Riseman, E.M., Hanson, A.R., Rosenfeld, A., "An Integrated Image Understanding Benchmark for Parallel Processors" (Invited Paper), *Journal of Parallel and Distributed Computing,* Vol. 11, pp. 1-24, 1990.

Weems, C.C., Hanson, A.R., Riseman, E.M., Rosenfeld, A., "IU Parallel Processing Benchmark", *Proc. CVPR,* Ann Arbor, MI, June 6-9, 1988, pp. 673-689.

Williams, L. R., (1990). "Perceptual Organization of Occluding Contours," *Proc. of DARPA IUW,* Pittsburgh, PA, pp. 639-649.