A KNOWLEDGE-BASED INTERACTIVE ROBOT-VISION
SYSTEM

by

T. Vámos - M. Báthor - L. Mérő - A. Siegler

Computer and Automation Institute, Hungarian
Academy of Sciences
Budapest, XI., Kende u. 13-17., 1111 Hungary

Abstract

A robot-vision project is reported which incorporates several existing AI methods and some new results. The ambition of the project is a system which can economically complete various intelligent tasks within the scope of mini- and microcomputers. The tuned composition of the applied methods provides a new and powerful approach to R and D engineering and workshop-operation.

1. INTRODUCTION

This paper reports on a robot-vision system. A lot of its features are similar to same existing systems, however, some of its characteristics are worthy of attention:

- a clear separation of four intelligence levels:
 - system Research and Development /R and D/
 - system Design and implementation
 - task teaching and its adjustment
 - production runs;
- a man-machine communication which controls the whole process and uses a real-time, simple graphic representation with special features for this kind of tasks;
- new real-time picture processing algorithms for contour following;
- a knowledge-based combination of picture processing and grammatical recognition methods;
- a 2D-3D inference mechanism;
- a well-performing system and application software.

2. GENERAL SYSTEM-PHILOSOPHY - HIERARCHY OF INTELLIGENCE

The goal of the project was a system which can complete various intelligent tasks economically, i. e. a system within the scope of mini- and microcomputers. This creates an optimal trade-off between the special task-oriented design and general problem solving. A feasible solution is hierarchical: elaboration of a very general methodology and a set of tools for a wide variety of problems combined by a shorthand design method for choosing the small subset of methods for a specific task, all based on an interactive system. Let us give details.

The highest level of hierarchy is the Research and Development in our special laboratory, containing all standardized system design tools, i. e. a special operations system, system design language, a sophisticated editor and the interactive graphics for displaying each step of the visual recognition, i. e. the result of each program segment, simulation of the robot manipulation and the teaching of the system. The graphic interaction is described in Section 4.

The system design languages are supported by a macroprocessor and Pascal-like special developments. The operating system and editor are display-oriented and especially designed for interactive operation.

The second level is that of the system design. The goal of the project consists of creating an industrial workplace, the elaboration of the hardware configuration, selection and improvement of software, system simulation, debugging. This is a CAD activity mostly started in the R and D lab using all of its facilities.

The third level is the adjustment and teaching of a single special task for a given site, including the selection of an even smaller subset of the software, parametrization and experimentation. This is a typical workshop-engineering problem.

The fourth level is the operational level. On the fourth or the operational level the user has similar responsibilities to that of a computer operator, for example, supervising the actual job stream, e.g. an assembly operation, the selection of a specific part etc.

The hierarchy of the levels is such that no level can modify any of the levels above it. However each level can communicate with the system by adding its own heuristics to the problem-solver.

The system is aimed at serving primarily assembly and material handling tasks. Later the system should also be extended to carry out inspections /texture, dimensions, contours/. The main limitations are:

- the objects should have well defined contours and/or surfaces;
- the number of objects used in the recognition process should be

limited /not exceeding 10-15/;

- human interaction is always necessary.

The limitations refer to the economy of intelligence as mentioned above: the low 4 decimal dollar range, which is set by the robot-hardware prices /\$ 25-35.000/ and the economy of human replacement. This is a rather uncommon approach of AI, but more common in many other engineering problems: e.g. a systematic optimization of services within a certain price limit. This limit is rapidly changing with technological progress, although this change is much less rapid for a complete system /including sensors, I/O, software etc./ than for single components, but the time complexity of the problems increases at a much steeper rate. That means that a rationale for limitation the complexity and the most economical man-machine division of labor is a useful design criterion. By applying the philosophy of the intelligence and task-levels this optimization problem can be approximated gradually as each level requires different resources, so that a very general operating framework is given for a wide variety of problems and the final result can be simple, comprehensive and very restricted.

3. PICTURE PROCESSING

After several experiments with special optical inputs such as laserscan /23/, at present the system uses a standard closed loop TV camera which yields a 192 by 144 matrix of pixels. This device was the result of a recent economic trade-off between input and processing hardware. 16 grey levels are used, the level-scaling is adjustable, each level can be turned on or off separately. The full picture can be windowed, zoomed and stored in memory. These functions are realized by a special hardware-unit. The preprocessing algorithm covers the picture by overlapping windows. The Mr-Vassy operator /11/, which is a simplified and faster version of the Hueckel operator, fits an optimal line or a stroke in each window. The local operator uses only two linear templates instead of the 8 used by Hueckel; if the pictures are not too noisy, the fit serves as an adequate basis for a future procedure /Fig. 3/. The goodness of the fit is measured by a statistical estimator and this value is later used as a weight of the stroke. At this point the algorithm can branch depending on the special application. The

strokes themselves can be used as input data of the common brute-force qualitative comparison methods /momenta, area/perimeter etc./. This first approximation can accelerate the succeeding steps.

The rather hairy picture of the strokes is elaborated by looking for optimal paths between hypothetical branching points where 3 or more homogeneous regions meet. A branching point is supposed in a region where the variance of the slopes of the strokes attains a local maximum.

Some arrays of strokes are assembled, each beginning and ending at hypothetical branching points. These arrays of strokes are called streaks. Some mathematical criteria are stated on the streaks expressing that the streaks are to follow the contour lines in an optimal way and that each contour line should be followed by just one streak /Fig. 4/. The streak assembling algorithm can be proved to satisfy these mathematical criteria. The erroneous hypothetical branching points can be corrected or neglected by this algorithm.

The algorithm extracts lines from the streaks by uniting the approximately identically directed consecutive strokes by comparing the slope-differences. This interpretation is generally weak, due to the uncertainty of the whole process, thus a probability value is assigned to each possible interpretation - while each streak may have various interpretations /Fig. 5/ - and this serves as an ordering of hypotheses in the linguistic recognition /Section 5/.

4. MODELLING OBJECTS - GRAPHIC INTERACTION

Recognition of objects, location, orientation and their manipulation - all need a representation of geometrical knowledge which should be optimal from the point of view of

- human interaction /simulation, programming aid, debugging, visual control/;
- display-manipulation, computation, memory requirements;
- 2D-3D conversion;
- access to the numerically expressed features realized by different programs /similar or easily convertible data structures/.

Considering these, a 3D-model data base was chosen for the recogni-

tion. Comparisons of the input pictures are made with computed 2D views. This approach avoids the 2D-3D restoration and makes use of the rotational and fast hidden-line computational abilities of the system.

According to the recognition procedure only linear and circular line sections are defined, the surfaces are planar, spheroidal, conical and cylindrical, although other second order surfaces are not excluded. The surface boundaries are real or imaginary edges, the latter separating "smooth" connections, visible and invisible parts as well as contours. The surfaces are directed, this is an important aid to the hidden-line algorithm /Fig. 7-10/. The data structure of the model contains not only the numerical geometric components but also additional information used for manipulation /grasping surface, orientation, grasping force etc./.

Teaching of the system is accomplished by interactive graphical programming. Although a direct input from the camera is available even in this phase, a human model building looks more straightforward. The available procedure is the following:

- menu-driven display of the numbered standard line and surface elements and stored complex or parametrized representations;
- human line-drawing and automatic correction in terms of the standard elements;
- pointing at picture elements.

An effective device for the robot recognition and manipulation is a development of the homogeneous coordinates concept /14/, a 4D augmented representation of the 3D objects, which helps in easy calculation of different views, rotation, linear transformations applying simple matrix multiplication. This method also serves for model building on the screen and at the same time in the data base.

Complex objects are formed by unification - the intersections are not calculated, because of the high computational requirements. For everyday problems of robotics this generally can be omitted, and if not, an easy manual intervention helps.

The hidden-line problem is essential in producing the 2D views for

comparison with the TV pictures. This algorithm is very fast, it always rotates the wire-frame model in the positive direction with respect to the z /vertical/ axis, calculates the contour edges of curved surfaces and by taking the orientations into account selects the "forward" and "back" surfaces and ignores the latter in cases where permitted, for example closed polyhedra bounded by convex polygons. The intersections of the edges are systematically ordered by the measure of hiding and checked logically for visibility. The algorithm is a modification of Loutrel's /8/.

5. RECOGNITION

The recognition process finds the highest probability fit between the 2D picture received from the interpretations of the strokes and those which can be derived by different views of the available 3D models. The chosen method is a kind of grammatical search.

An available a priori knowledge consists of a set of faces /"countries"/ which appear in at least one of the 2D views of the 3D models and, in addition, all possible arrangements of these faces which can compose a meaningful picture in the scenery of the knowledge-base.

Thus at first all possible kinds of faces are assembled from the interpretations of the streaks. A streak in each face can be represented by only one of its interpretations. A reliability value of each face is computed from the probabilities of the interpretations of the streaks used in assembling that particular face.

All those combinations of the lines and arcs from the interpretations of the streaks which constitute meaningful faces, are assembled into possible faces. These planar configurations /e.g. parallelograms, triangles, loops etc./ are guessed by allowing some tolerance-thresholds and the reliability-value of the guess is computed. /Fig. 6/

Finally the possible faces are assembled into complete pictures /line drawings/. The a priori knowledge of the model data-base concerning the connections and relations of the faces is utilized

again, each meaningful 2D line drawing is tried to be assembled from the possible faces. The assembled variants are also provided with reliability-values derived from the reliability values of the constituting faces. The highest probability yields the supposed recognition, an identification of the 3D model and the TV picture /Fig. 11/. Symmetry can cause some ambiguities but this may be irrelevant or easily eliminated. Coarse evaluators or parameters can also be included in these final procedures as search accelerators.

The search procedures have been implemented by using V. Galló's /6/ parsing program.

The final result of the recognition process is a matrix which describes the transformation of the object from the model's coordinate system to its actual position. The size of the models are the same as the real size of the objects and the projection matrix of the input camera is also supposed to be known. Thus a best fitting matrix that transforms the 2D projection of the model into the input picture, can be computed from the one-to-one correspondences between the vertices of the two pictures. If there are several one-to-one correspondences, the matrix of realized least-square error will be accepted.

6. HARDWARE AND MANIPULATOR CONTROL

The visual input was shortly described under Section 3. The computer is a 16 bit, 40 KB mini /VIDEOTON R-10 - a licence of the French CII-Mitra 15/ equipped with a minidisc, magtape unit, graphic display. The manipulator has two versions: an orthogonal and a polar one /Fig. 12/, both 6 degrees of freedom plus the grasp movement, force and torque feedback, similar to the Draper Lab solution /12/. The computer controls the robot via a 3D CNC equipment /17/ simultaneously in all degrees of freedom. Important features of the robot-control:

- it can be simulated and controlled on graphic display;
- it can be taught by manual operation or by graphic simulation;
- it uses the same spatial transformation algorithms as the model building and recognition, i. e. the relative positions due to
 * object location

- * arm movement

- * visual recognition errors etc.

can be calculated quickly and effectively.

7. TIME REQUIREMENTS

The whole recognition procedure takes about 30-50 seconds for an object shown e.g. on Fig. 2. The times of the particular algorithms are as follows:

The hidden line algorithm is extremely fast, e.g. it produced Figs. 10, 11 within 2 seconds each. The edge detection operator takes approx. 4 seconds for a TV input picture. The algorithm which assembles the strokes into streaks requires 10-15 seconds. The grammatical searching procedure to find the possible faces and the possible object drawings from the streaks takes 5-8 seconds. The other on-line algorithms take less than 1 or 2 seconds altogether.

All the times listed are experimental results on the R-10 minicomputer with a 2,5 μ s add-time. These time data enable this system to meet on-line requirements on a faster computer and possibly make use of some parallel processing.

8. APPLICATIONS

The system is primarily assembly-oriented. This task seemed to be a good paradigm of the problem and in fact, till now it has been handled in this way. It means that only laboratory environment tasks have been solved and a system as a mass-produced workshop-robot will not be available for the next two years. Less ambitious applications of the operating results are being realized recently: a recognition system in a bus-body factory identifies the metal sheets for controlling a painter robot, and an application in neurobiological research. The latter is an interactive system which should detect neuronal nets in microscope-sections.

9. FUTURE TRENDS

Our goal of paramount importance is how to collect enough experience with the complete laboratory system to enable research and the in-

dustry to make a real-life environment design. We are well aware of the many problems that naturally arise in similar projects. E.g. illumination - how to use different light sources in solving the trouble-making shadow and shine effects by applying the knowledge-base in a highly intelligent way. Robot control and pattern recognition are centralized in one computer. Our future plans include considerations, whether to use distributed systems. Some important components realized in the new CNC-system and graphic display are already available thanks to the achievements of other groups of the Institute.

Especially in Section 5 we described our approach to the problem of similarity and distances. Distance is a very common notion in statistical pattern recognition. The metrics - which is the decisive part of the problem - lies in the very statistical nature of the approach; a good metrics is such that it can discriminate the clusters with the greatest reliability. By this way a good statistics provides an optimal metrics. In our case the metrics is more logical than statistical, being bound to the individual object which should be identified for the individual task. What to consider similar, which details to suppose identical /e.g. three straight lines or one arc/ are task dependent. Some approaches for a general quantification of similarity or distance in this logical sense are known /5, 15, 18/ but these methods are applicable only in a priori evaluable situations. In the knowledge-based interactive system most measures of distance should be taught and experimented. The crucial features can be very diverse /one dimension, one certain detail, a complex interrelation of characteristics/, their selection is within the scope of human experience. We think that the best solution is to provide the human operator with interactive linguistic and graphic tools so that he can find these distances /if they are not well-known before/ and communicate them to the system.

The above ideas should be outlined and elaborated much better during the next experimental period.

CONCLUSIONS

A knowledge-based man-machine intelligent robot system is reported

having special features in fast real-time pattern recognition, teaching, modelling and control, which makes use of graphic interaction. A very general industrial problem solver is outlined with a hierarchy from R and D level to the operational level. The limitation of intelligence is a practical, economical trade-off which provides a variable optimum for the man-machine distribution of tasks.

References

- /1/ A.P.Ambler, H.G.Barrow, R.M.Burstall, R.J.Poppelstone: A Versatile Computer-Controlled Assembly System. Proc. 3rd IJCAI /Stanford/, 1973, pp. 298-307
- /2/ M.Báthor: Interactive Picture Manipulation. 2nd Hungarian Computer Science Conference /Budapest/, 1977, pp. 168-177
- /3/ M.Báthor: Hidden-Line Algorithm for a Robot Experiment. Ph.D. Thesis /Budapest/, 1977 /in Hungarian/
- /4/ B.G.Baumgart: Geometric Modelling for Computer Vision. Stanford Memo AIM-249, 1974
- /5/ K.S.Fu: Stochastic Tree Languages and Their Applications to Picture Processing. International Symposium on Multivariable Analysis /Pittsburgh/, 1978
- /6/ V.Galló: A Program for Grammatical Pattern Recognition. 4th IJCAI /Tbilisi/, 1975, pp. 628-634
- /7/ V.Galló: Sistema dlya obrobotki spiskov dlya intelligentnovo robota. 2nd Hungarian Computer Science Conference /Budapest/, 1977, pp. 400-411 /in Russian/
- /8/ P.P.Loutrel: A Solution to the Hidden-Line Problem for Computer-Drawn Polyhedra. IEEE Trans. Comp., C-19, 1970, pp. 205-213
- /9/ L.Mérő: A Quasi-Parallel Contour Following Algorithm. Proc. AISBIGI Conf. on AI /Hamburg/, 1978
- /10/ L.Mérő, T.Vámos: Real-Time Edge Detection Using Local Operators. 3rd IJCPR /Coronado/, 1976, pp. 31-36
- /11/ L.Mérő, Z.Vassy: A Simplified and Fast Version of the Hueckel Operator. 4th IJCAI /Tbilisi/, 1975, pp. 650-655
- /12/ J.L.Nevins et al.: Exploratory Research in Industrial Modular Assembly /Cambridge, Mass./, 1977
- /13/ M.Potmesil: An Implementation of the Loutrel Hidden-Line Algorithm. Rensselaer Polytechn. Inst., TR CRL-49 /Troy, N.Y./, 1976
- /14/ L.G.Roberts: Machine Perception of 3D Solids. MIT Press /Cambridge, Mass./, 1965, pp. 159-197
- /15/ A.Rosenfeld, R.A.Hummel, S.W.Zucker: Scene Labelling by Relaxation Operations. IEEE Trans. SMC-6, 1976, pp. 420-433
- /16/ Y.Shirai: Analyzing Intensity Arrays Using Knowledge about Scenes, in the Psychology of Computer Vision /ed.P.H.Winston/ /New York/, 1975
- /17/ A. Siegler: Computer Controlled Object Manipulation. 2nd Hungarian Computer Science Conference /Budapest/, 1977, pp. 724-738

- /18/ E.Tanaka, K.S.Fu: Error-Correcting Parsers for Formal Languages. IEEE Trans. Comp. C-27, 1978, pp. 605-616
- /19/ T.Vámos: Industrial Objects and Machine Parts Recognition, in Applications of Syntactic Pattern Recognition /ed. K.S.Fu/, /Heidelberg/, 1977, pp. 243-267
- /20/ T.Vámos: CAD-Marriage with AI Methods, Views Based on Dual Experiments. Prepr. of IFIP WG 5.2 Conf. on AI and PR in CAD /Grenoble/, 1978, Session 5
- /21/ T.Vámos: Automatic Control and Artificial Intelligence /invited survey/. Prepr. of 7th IFAC World Congress /Helsinki/, 1978, 4, pp. 2355-2369
- /22/ T.Vámos, Z.Vassy: Industrial Pattern Recognition Experiment - A Syntax Aided Approach. Proc. 1st IJCPR /Washington/, 1973, pp. 445-452
- /23/ T.Vámos, Z.Vassy: The Budapest Robot - Pragmatic Intelligence. Proc. of 6th IFAC World Congress /Boston/, 1975, Part IV/D, 63.1

Fig. 1: An object in the scene

Fig. 2: Digitized input

Fig. 3: The strokes found in the picture

Fig. 4: The streaks of strokes with the branching points found

Fig. 5: All possible interpretations of the streaks

Fig. 6: The lines from the streaks used in the recognition

Fig. 7: 3D wire-frame model
of the object

Fig. 8: The model being built
(hidden line version)

Fig. 9: The model, some surfaces
are not defined yet

Fig. 10: A projection of the
perfect model

Fig. 11: Another projection to
be matched to Fig. 6

Fig. 12: The robot arm