COMPUTER VISION, GRAPHICS, AND IMAGE PROCESSING 36, 372–386 (1986)

Computer Vision Theory: The Lack Thereof

ROBERT M. HARALICK

Department of Electrical Engineering, University of Washington, Seattle, Washington 98195

Received May 7, 1986

The first part of the paper presents a brief description of how science and the scientific method might apply to computer vision and then contrasts this perspective with three strong papers in the pose detection area of computer vision. The contrast illuminates the shortcomings and the lack of a completed computer vision theory in these papers. The second part of the paper outlines a theoretical approach and problem statement to 3D object matching through a sensor projection, a Bayesian approach to the robust estimation of camera parameters, and to the hypothesis verification problem. © 1986 Academic Press, Inc.

I. VIEWPOINT

Scientific advancement proceeds on two frontiers: the experimental frontier and the theoretical frontier. In the experimental frontier researchers perform some combination of exploratory experimental work and formal hypothesis testing. In exploratory experimental work experiments are performed and data is gathered in the hopes that some pattern in the observed data can be deciphered which would suggest a formal hypothesis to test. In the hypothesis testing mode, the experiments being performed are done by explicitly setting up some controlled situation and testing if the resulting observations agree with what one would expect to observe if the hypothesis were true. The hypothesis could come from a conjecture, a law which follows from a theory, or a hypothesis test which attempts to replicate the results of a previously reported experiment. In the theoretical frontier, researchers perform some combination of synthesizing experimental data and existing theory into a more comprehensive coherent and general theory. The language of the theory is expressed in a mathematical form in all the hard sciences.

As a science, computer vision has its experimental and theoretical aspects. In the theory of the science of computer vision one would expect to find the laws and principles by which computer algorithms can be designed to solve a variety of vision tasks from industrial inspection, robot assembly, autonomous vehicle navigation, and general 3-dimensional scene understanding. In the experimental results reported in the archival scientific literature for computer vision, one would expect to find clear descriptions of controlled situations under which the experiments are performed, a precise statement of the algorithm being used, and a statement of the results which includes some measure of the certainty of the stated results. In the theoretical results reported in the archival scientific literature one would expect to find a variety of partial or incomplete theories each of which gives a precise statement of the particular computer vision problem the theory addresses. The content of the theory would develop a set of laws, principles, and associated algorithms which logically proceed from the initial problem statement and assumptions pertaining to the reality the theory addresses. The algorithms would accept for input an appropriate image or images and perform a calculation which provides an

0734-189X/86 $3.00

answer which is correct modulo the amount of noise in the data and the adequacy of the theory.

When a broad examination is made of computer vision research, it becomes apparent that the science is young and immature. The pockets of theory are sparse. The amount of replication is nearly non-existent for the complex algorithms. Very few experiments are reported on enough image data so that the certainty of the results can be stated. There appears to be no agreement on controlled data sets for any experiments. Many experiments have a "Look ma, no hands" aspect to them. Indeed, the experiments are often extremely complicated because the computer vision algorithms are extremely complicated. The algorithms are so complex that the algorithm details often cannot be entirely reported in a single paper. Not only is a precise statement of the problem difficult, but it is nearly impossible to find the appropriate assumptions which make the mathematical derivations which proceed from the problem statement simultaneously mathematically tractable and a reasonable description of reality.

Upon reading the literature, one even gets the feeling that perhaps the algorithm itself is sufficient, without a statement of what problem is being solved or without a statement of the degree to which any problem is being solved. At the 1985 NSF-sponsored workshop on model-based computer vision held in Orlando, some participants even held the view that it was not important to try to state problems precisely or attempt to derive from formal problem statements optimal, or near optimal, solutions. Those people held that the only thing which matters is whether the algorithm produces reasonably good results in the application for which it is intended.

II. WHY?

Why does this state of affairs exist? Is it because computer vision is young and the problem is difficult? Certainly so, but this answer is a quick one and not a helpful one in understanding the shortcomings of the field. In this section we attempt a deeper understanding of why.

Every science develops a body of principles which are used in solving the application problems to which the science is directed. To be sure, the body of principles undergoes refinement and changes as the science develops. For computer vision, the body of principles would contain the problem statement and solution techniques for a variety of computer vision problems. These problem statements would be statements of canonical form problems. For example, in pattern recognition under a class conditional Gaussian assumption, the quadratic form of the optimal maximum likelihood decision rule is well known. In computer vision, the canonical problem of relational matching by a relational homomorphism is well stated and efficient tree search techniques for the computation of the matching function are well known. In numerical analysis, stable techniques for performing the singular value decomposition of a matrix or for determining the eigenvectors of a matrix are well known and are available in standard software packages such as LINPACK, EISPACK, and SPSS.

Computer vision, to be sure, has its current bag of tools. But it is not the case that very many of these tools constitute the optimal solution technique to a well defined problem. The thesis of this paper is that computer vision will advance if more effort

were put into the definition of canonical computer vision subproblems and if more effort were put into their optimal solution.

III. POSE DETECTION

To illustrate the points raised in Sections I and II in a more concrete way, we consider the problem of object pose detection. We will draw upon three strong and solid papers in this area, papers by Roberts [4], Perkins [3], and Stockman, Kopstein, and Benett [5]. These papers represent work of which the authors as well as the computer vision research community can be proud. First we will give a brief summary of the work and then discuss some of its shortcomings relative to the definition of subproblems and to their optimal solution.

Object pose detection is the process by which a 2-dimensional perspective projection of a 3D object is analyzed to determine the object's pose (position and orientation) with respect to a given 3D coordinate axis system. The analysis first depends on the capability of matching the 2D perspective projection of the object on the image to a 3D model of the object. Then given the match, the unknown parameters of the perspective projection must be determined. A transformation of coordinates to the given 3D coordinate system can then produce the object's pose.

The first paper to explore a solution to this problem was by Roberts [4] who matched topologically equivalent points. Roberts [4] and Duda and Hart [1] give a least squares solution to the determination of the unknown projective geometry parameters once the match points are known. Perkins [3] discusses a solution to the 2D matching problem as well as to the problem of determining the 2D projection transformation parameters. Stockman, Kopstein, and Benett [5] also match similar 2D structures (vectors) from the image to the model. Each match determines a transformation. The set of transformations is clustered and the cluster having the largest number of transformations is the one chosen as the one defining the most likely projective transformation.

Perkins [3] first represents a 2D shape by the chain of its boundary points where each boundary point has a tangent direction which is produced by the edge direction of the edge operator detecting the boundary point. To determine straight line segments or circular arc segments each boundary sequence must be broken at points of high curvature change. Perkins computes curvature at each boundary pixel as the ratio of the difference in tangent angle of adjacent boundary points to the difference in position of the adjacent boundary points. Then segments of boundary pixels are fit to a straight line or to a circle. The resulting arcs are used in the model matching.

Model matching proceeds by cross correlating a tangent angle vs arc length curve of the model against a tangent angle vs arc length curve from a detected boundary point sequence of the observed object. The position determined by the shift producing the highest correlation and the tangent angle at that position constitute sufficient information to determine the translation and rotation angle of the unknown sensor projection.

Stockman, Kopstein, and Benett [5] perform a local feature detection on the 2D image and 2D model. Matching takes place based on structures which are determined by pairs of features which can stand in a legal spatial relationship. A structure from the image matched to a structure in the model determines the unknown 2D rotation and translation parameters. Thus to every possible match

there is a corresponding rotation and translation vector. These vectors are clustered. The representative vector from the strongest cluster determines the estimate of the unknown transformation.

III.1. The Shortcomings

There are a variety of shortcomings, much easier to see on hindsight, and we will discuss only a few. In the seminal work of Roberts [4], once a set of image and model-matched points are obtained, the solution to the parameters of the unknown perspective projection is done in an equal-weighted least squares sense on the entries of the 4×4 perspective projection matrix. This perspective projection matrix has 16 entries which depend on the 6 unknown parameters (3 for rotation and 3 for translation). The least squares solution, however, is done without regard to the independent constraints which are known to hold. Furthermore, the solution technique is a least square solutions technique. The equal weight least squares solution is optimal only in the case where the errors of the observations in fact are independent and identically Gaussian distributed. Should the error distribution be thicker tailed than a Gaussian or should there be outliers, the least squares solution is the technique with least virtue. Vision needs robustness. So for the Roberts perspective projection parameters case, the wrong problem was posed and the solution technique given is the right technique for an incorrect reality.

Perkins [3] put a variety of techniques together in a way which handled some relatively hard vision tasks. The paper is innovative, solid, and representive of the consistently good work which Perkins does. The Perkins paper, however, is an example of the "look Ma, no hands" approach. Consider, for example, the arc segmentation problem discussed in the Perkins paper. A sequence of points is given and each point is tagged with a noisy observation of its tangent angle. It is known that the sequence of points is sampled from an arc having only straight line and circular arc segments, each segment being greater than some minimum length. The problem is to identify each segment. Identification means estimating the starting and stopping points for each segment, classifying the segment as being a straight line segment or circular arc segment, and estimating the free parameters in whatever is the most appropriate parametric form.

An optimal solution to this problem is one which assumes a model having some resemblance to reality and which under the assumed model produces an answer which is the most probable or one which has the greatest utility. Utility can be defined in terms of an accurate numerical result balanced with the amount of computational required to obtain the result. Suboptimal solutions might involve changing the actual problem to a problem close to the actual problem but one which is more mathematically or computationally tractable. Suboptimal solutions might involve using an approximation in the derivation of the algorithm.

What does Perkins do to solve this problem? He estimates curvature by a finite difference approach using the noisy tangent angle of the edge detector and ignoring the point position information inherent in the given sequence. He uses high curvature change, presumably also estimated by a finite difference, to determine segment end points. Under what model, if any, is this technique for estimating third derivatives an appropriate one? What experimental data has been gathered to determine the performance of this technique in situations where the noise, however

it is defined, becomes greater and greater. If the model were stated and the technique an appropriate one under the model, then I as a reader could think about whether the reality in which I need to perform this task is one consistent with the model. If the model is not stated, then other researchers will prefer their own applications to some one else's. And the proliferation of ad hoc computer vision techniques just continues.

If the model is not stated but experimental data were given which shows that the technique performs better than other techniques, that too would be useful. Certainly it is the case that sometimes intuition about a technique precedes its formal understanding. But in these cases, the researcher reporting the results is obliged to be empirically convincing. Without controlled experiments, standards, and comparisons the technique represents another ad hoc trick of the trade.

The Stockman et al. paper is probably the strongest of the three papers in so far as the robustness of the estimated transformation parameters are concerned. It is certainly ad hoc in the sense that clustering is ad hoc. Although the increased robustness of the estimates is intuitively obvious, one does not really know if it is the most robust over all techniques with the same computational complexity. One does not even know in what sense the technique might be a suboptimal technique. Had the Stockman paper shown the robustness theoretically, then as soon as I as a researcher commit myself to doing a vision task like the one Stockman did, I must, if I am rational, commit myself to using his technique if the computational resources I am willing to commit to the solution are the same as those he committed. But the paper does not prove the robustness. Hence, if I am a researcher trained in the artificial intelligence tradition, I read the paper, put another interesting tool in my tool bag, and wonder how I might develop a new and better ad hoc tool.

If I am trained in the artificial intelligence tradition, I certainly do not want to think about solving the problem the most optimal way under my favorite criteria of optimality because I know that once I can express the problem as any kind of an optimization problem it leaves the realm of artificial intelligence and becomes some kind of pattern recognition or mathematical problem which is repugnant to me and my colleagues. And so this ad hoc investigation continues to propagate.

In simple and concrete terms, what is this fuss all about? To do model-based vision, a common subtask in virtually all vision research, the estimation of the sensor transformation parameters is required. The data going into this estimation process is known to be noisy with outlier observations from foreign populations.

Some might suggest that this is the problem that the Fischler and Bolles [2] random consensus technique solves. Intuitively, the random consensus technique also has more robustness than least squares. Fischler and Bolles did do some experiments. But where is the comparison, experimentally or theoretically, which will guide me to select the Fischler and Bolles technique or the Stockman technique?

It would seem that at this point in the maturity of computer vision as a science, we would know how to best solve this problem. And the ad hoc situation in this instance where it is easier to precisely define the problem is surely better than the rest of computer vision where the problems may be more difficult to precisely define. Thus the thesis of this paper: Computer vision has little theory. It has much ad hoc research. Computer vision just has not reached the maturity of a hard science.

To do something about this we need to be sensitive to what is ad hoc, to what techniques are currently in fashion, and to what is science. We need to take a

conscious stand, develop this consciousness in our actions, and move toward the science of computer vision.

So in three good papers, papers which are, in my opinion, better than perhaps 95% of the archival papers, there are shortcomings. It is easy to be critical and find such shortcomings. But the purpose of this paper is not just to be critical. Its purpose is to raise the sensitively of computer vision researchers so that their future papers might have fewer methodological shortcomings.

It is also the purpose of the paper to practice the methodology preached. Although I try to practice the methodology, I am not able to necessarily practice it perfectly. But I do try. In Section IV, I attempt a formalization of the object matching problem through a sensor projection. The formalization suggests a solution procedure, one of whose subtasks is like the estimation of an unknown perspective projection parameters problem. An informal solution to this problem is also given. In Section V a more formal approach, statement, and solution of the robust estimation of perspective projection parameters is given. The solution technique of Section V motivates the computationally simpler but less optimal technique discussed in Section IV which is a generalization and refinement of the Fischler and Bolles technique. Finally, in Section VI one Bayesian approach is given to the hypothesis verification problem, a problem with which most vision control systems have to contend.

IV. 3D OBJECT MATCHING THROUGH A SENSOR PROJECTION

Most image sensors such as optical, infrared, radar, sonar, and X rays produce some kind of a 2D projection of the 3D objects viewed. To structurally recognize a 3D object from a 2D view taken from an unknown position requires a relational matching of the object with the image of the object. The difficulty of the matching process is that some of the relationships which might hold for the 3D object can be lost in the 2D image of the object. Some relationships which might hold for a 2D image of an object may not hold for the 3D object itself. Thus a total relational matching cannot be counted on to solve the matching problem. What is required is a relational matching that can operate through a sensor projection. The relational matching must be able to proceed using whatever subset of features which appears on the image and match these into the 3D object model. The match between image and object model must satisfy both the relational constraints and the sensor geometry projection equations.

We assume the physics governing each type of sensing mode is known. Given the sensor type and its position $t = (t_x, t_y, t_z)$ and orientation $\theta = (\theta_x, \theta_y, \theta_z)$ in the object coordinate system, the sensor geometry projection equations describe how each point (x, y, z) in the 3D world is transformed to a point (x', y') on the 2D image. Let these equations by specified by

$$x' = f(x, y, z; t, \theta)$$
$$y' = g(x, y, z; t, \theta).$$

The functions f and g are many to one. Given a 2D position on the image, there are many corresponding 3D points. However, if there are N points on the image (N being about 4 for most sensors) for which the corresponding positions of the N 3D points are known, then it is possible to solve the geometry equations for the position

$t = (t_x, t_y, t_z)$ and orientation $\theta = (\theta_x, \theta_y, \theta_z)$ of the sensor. It is this fact which permits a relational match to proceed through a sensor projection.

Matching determines a correspondence between image points and object points which are simultaneously relationally consistent and satisfy the sensor geometry projection equations. The eligible points to be considered on the image must be points which are easily distinguishable. Such points can be the center of bright spots, the center of dark spots, line end points, corners between two lines, or corners between two edge arcs. These points can be detected by some low-level neighborhood feature extraction operation on the image.

We denote the set of these distinguished image points by U. Associated with each point u is its image coordinates $x'(u)$ and $y'(u)$, and the type of point or label of point $P(u)$. The label set for points is denoted by L which can be defined as, for example, $L = \{$*bright spot*, *dark spot*, *line corner*, *edge corner*$\}$. The point labeling function P has domain U and range L; P: $U \rightarrow L$.

The distinguished points stand in relation to one another through other detected features such as edge arcs, line arcs, or containment in the same homogeneous region. We denote by A the set of relations which can exist between image points. For example, A can consist of

$$A = \{\textit{straight arc edge}, \textit{curved arc edge}, \textit{straight arc line}, \textit{curved arc line}, \textit{containment in same homogeneous region}\}$$

Pairs of distinguished points from U tagged with a relation label from A define the relational contraints. We denote this relation R; $R \subseteq U \times U \times A$. If $(u_1, u_2, a) \in R$, then points $u_1$and u_2 stand in relation a on the image. The total relational structure on the image is then given by the 5-tuple (U, L, A, P, R).

The distinguished points on the image come from corresponding points on the 3D object being imaged. For a fixed object, let V be the set of 3D points which can give rise to the detected 2D feature points. Associated with each point $v \in V$ is its 3D coordinates $(x(v), y(v), z(v))$, which are the coordinates relative to the object frame. Also associated with each point $v \in V$ is a set of types of distinguished feature points it can give rise to on the image. We denote by $Q \subseteq V \times L$ the relation associating with each point in V the types of point labels its corresponding point on the image can be. Q, although a relation rather than a function, has a role in the object's relational structure which is analogous to the role P has in the image relational structure.

Depending on sensing modality, object shape, and viewing direction, pairs of points in V (on the object) can have corresponding pairs of points in U (on the image) related by one of the relations in A. For a fixed sensing modality and object, we let $S \subseteq V \times V \times A$ denote the set of all triples (v_1, v_2, a) such that for some viewing directions of interest object points, v_1 and v_2 give rise to image points which stand in relation a. The relational structure of an object is then given by the 5-tuple (V, L, A, Q, S).

Model based object matching through a sensor projection can be defined in terms of relational consistency between the image relational structure (U, L, A, P, R) and the model relational structure (V, L, A, Q, S) combined with the sensor projection geometry consistency between the points in U which match to points in V. The matching procedure must determine some maximal part of (U, L, A, P, R) which matches into (V, L, A, Q, S) and which is consistent with the projection equations.

The matching inevitably involves a search and the difficulty is how to make that search be over the smallest set of alternatives possible.

To make the search efficient, we break the problem up into two parts. First we search over all subsets U' of distinguished points in the set U. We can structure the search to involve only subsets of size N where N is the smallest number of 2D image points which when matched with 3D object points can guarantee a solution for the unknown position and orientation of the sensor. If prior information is available that the points on the image must be from a particular subset U_p of U, then the search only selects U' from within the known subset U_p. Having selected a subset U', we perform a search to determine all relational matches. A function h: $U' \to V$ is a relational match if and only if

(1) corresponding points are of the same type, that is $u \in U'$ implies $(h(u), P(u)) \in Q$

(2) pairs of corresponding points have the same relational label, that is $R \circ h \subseteq S$, where $R \circ h$ is defined by

$$R \circ h = \{(v_1, v_2, a) \in V \times V \times A \mid \text{for some } (u_1, u_2, a) \in R,$$
$$v_1 = h(u_1) \text{ and } v_2 = h(u_2)\}.$$

If prior information is available that certain points $u \in U$ can only match to certain points $v \in V$, then the search for the relational match h is restricted. That is, if $H \subseteq U \times V$ is the prior contraint, then the search for h is restricted to all functions in $H \cap (U \times V)$. After the search has established a candidate relational match, it must be verified. The verification has two conditions. The first condition established that there exists a sensor position $t = (t_x, t_y, t_z)$ and sensor orientation $\theta = (\theta_x, \theta_y, \theta_z)$ by which every 3D object point from V with coordinates (x, y, z) corresponding to a distinguished 2D image point from U' with coordinates (x', y') satisfies the sensor geometry projection equations. That is, we must verify there exists a t and θ such that for every $u \in U'$,

$$x'(u) = f(x(v), y(v), z(v); t, \theta)$$
$$y'(u) = g(x(v), y(v), z(v); t, \theta)$$

where $v = h(u)$. Establishing the existence of such a t and θ is done by solving the above set of sensor geometry projection equation for t and θ. If no position and orientation can be found which makes the sensor geometry projection equations satisfied, then the candidate relational match is not a valid 2D to 3D correspondence.

If a position and orientation can be found which makes the sensor geometry projection equations be satisfied for the match h, then the second part of the matching procedure seeks to find confirming information for the correspondence. Confirmation involves finding additional 2D points and relations, not used in the initial candidate matching, which simultaneously have corresponding 3D points which satisfy the sensor geometry projection equations and the relational match. The confirmation phase of the matching process is different from the candidate search phase of the matching processing in that because the position and orientation

of the sensor is known, there need be no search to establish confirmation. There need be only some checking.

In the confirmation phase, each 3D point $v \in V$ has 3D coordinates $(x(v), y(v), z(v))$. Sensor position t and orientation θ are known. By the sensor geometry projection equations, the corresponding 2D image point has position (x^*, y^*) given by

$$x^* = f(x(v), y(v), z(v); t, \theta)$$
$$y^* = g(x(v), y(v), z(v); t, \theta)$$

Either the point (x^*, y^*) is the position of a distinguished feature point from U or not. If it is a distinguished point, then there exists a $u \in H$ such that $(x^*, y^*) = (x'(u), y'(u))$. The set U^* defined by

$$U^* = \{u \in U | \text{for some } v \in V, x'(u) = f(x(v), y(v), z(v); t, \theta)$$
$$y'(u) = g(x(v), y(v), z(v); t, \theta)\}$$

is the set of all points on the 2D image which can be used for confirmation. The set

$$V^* = \{v \in V | \text{for some } u \in U^*, x'(u) = f(x(v), y(v), z(v); t, \theta)$$
$$y'(u) = g(x(v), y(v), z(v); t, \theta)\}$$

is the set of all points on the 3D object which can be used for confirmation. The function h^*: $U^* \to V^*$ defined by

$$h^*(u) = v \quad \text{if} \begin{cases} x'(u) = f(x(v), y(v), z(v); t, \theta), \\ y'(u) = f(x(v), y(v), z(v); t, \theta), \\ (v, P(u)) \in Q, \end{cases}$$

establishes all possible candidate matches between points on the image and points on the object which are consistent with the sensor geometry projection equations and have the same permissible type labels. Computing h^* is direct once t and θ are known. For each 3D point $v \in V$ determine its 2D projection and see if at the projection position there exists a 2D point $u \in U$. If so check their type labels. If they both have the possibility of same type labels then add the match to h^*.

The function h^* is essentially produced by propagating forward the constraints implied by the candidate match function h. In this sense its generation is analogous to the forward checking procedure used in constraint satisfaction tree searches [6].

Having computed h^*, the confirmation process can be completed. The function h^* necessarily includes the candidate matching function h. Confirming evidence can in part be given by the difference in size between h^* and h; that is $\#h^* - \#h$. Confirming evidence can also in part be given by the number of relational matches h^* can establish beyond that which h established. This number is given by

$$\#[(R \circ h^*) \cap S] - \#[(R \circ h) \cap S].$$

The next section suggests other criteria motivated by a Bayes approach.

If the function h^* significantly extends the match h on the number of relations matched then confirming evidence can be considered to have been provided and the extended candidate match h^* becomes the match between object and image.

V. A. BAYESIAN APPROACH TO ROBUST ESTIMATION OF CAMERA PARAMETERS

V.1. Problem Statement

Suppose $b_1, \ldots, b_N$ are known points in a 3D space whose observed perspective projections are $x_1, \ldots, x_N$. The position, rotation, and focal length of the camera, i.e., the parameters which determine the perspective projection, are not known. For any value of camera parameters a and any 3D point b_n, the function μ determines the ideal perspective projection $\mu(a, b_n)$ of b_n. An observed perspective projection x_n of b_n is a noisy instance of $\mu(a, b_n)$ with probability q. With probability $1 - q$, x_n comes from $U(-Large, Large)$. Such an x_n represents a contaminating measurement. For $n \neq m$, x_n is independent of x_m. The problem is to estimate a.

V.2. Analysis

Let $y_1, \ldots, y_n$ be independent random variables characterized by

$$y_n = \begin{cases} 1 & \text{with probability } q, \\ 0 & \text{with probability } 1 - q. \end{cases}$$

The y's are independent of x_n's. When $y_n = 1$, the observation x_n is not an outlier. When $y_n = 0$, the observation x_n is an outlier coming from a foreign population. We wish to find the most probable value of a, the unknown parameters given the observed and known information,

$$\begin{aligned} &P(a|x_1, \ldots, x_N, b_1, \ldots, b_N) \\ &= \frac{P(x_1, \ldots, x_N|a, b_1, \ldots, b_N)P(a, b_1, \ldots, b_N)}{P(x_1, \ldots, x_N, b_1, \ldots, b_N)} \\ &= \sum_{y_1, \ldots, y_N} \frac{P(x_1, \ldots, x_N, y_1, \ldots, y_N|a, b_1, \ldots, b_N)P(a|b_1, \ldots, b_N)P(b_1, \ldots, b_N)}{P(x_1, \ldots, x_N, b_1, \ldots, b_N)}. \end{aligned}$$

Because of the conditional independence of x_n, y_n on a and b_n, and because of the independence of a from $b_1, \ldots, b_N$,

$$P(a|x_1, \ldots, x_N, b_1, \ldots, b_N) = \sum_{y_1, \ldots, y_N} \frac{\left[\prod_n P(x_n, y_n|a, b_n)\right]P(a)}{P(x_1, \ldots, x_N|b_1, \ldots, b_N)}.$$

Rewriting $P(x_n, y_n|a, b_n)$ there results,

$$P(a|x_1, \ldots, x_n, b_1, \ldots, b_N) = \sum_{y_1, \ldots, y_N} \frac{\left[\prod_n P(x_n|a, b_n, y_n)P(y_n|a, b_n)\right]P(a)}{P(x_1, \ldots, x_N|b_1, \ldots, b_N)}.$$

But y_n is independent of a and b_n. Hence

$$P(a|x_1, \ldots, x_N, b_1, \ldots, b_N) = \sum_{y_1, \ldots, y_N} \frac{\left[\prod_n P(x_n|a, b_n, y_n)P(y_n)\right]P(a)}{P(x_1, \ldots, x_N|b_1, \ldots, b_N)}.$$

From the above equation, the value of a achieving the maximization is a computationally intensive task because the summation over all possible values $y_1, \ldots, y_N$ involves 2^N terms. The difficulty is not so much that there are 2^N terms to be summed, but that the maximization to be performed is much more complex when the expression to be maximized has 2^N terms to be summed, each term being a product of N functions.

However, there is a condition under which things can be considerably simplified. The simplifying condition amounts to assuming that if $y_n = 1$, indicating that we are conditioning under the assumption that x_n is not an outlier, then the probability density $P(x_n|a, b_n, y_n = 1)$ is peaked. Hence if the observation x_n is really not an outlier $P(x_n|a, b_n, y_n = 1)$ will be rather high. We take rather high to be a probability density greater than unity. However, if the observation x_n really is an outlier, then $P(x_n|a, b_n, y_n = 1)$ will be rather small. We take rather small to mean rather small in comparison to $((1 - q)/q)P(x_n|a, b_n, y_n = 0)$. The reason for this will be obvious shortly. Since

$$P(x_n|a, b_n, y_n = 0) = \frac{1}{2 \text{ Large}} = \varepsilon,$$

this means that when x_n is an outlier $P(x_n|a, b_n, y_n = 1)$ will be small in comparison to $((1 - q)/q)\varepsilon$.

The motivation for these comparisons is as follows. Let G^* be the set of indexes of observations which are really not outliers and B^* be the set of indexes of observations which are really outliers. For any one of the 2^N values of $y_1, \ldots, y_N$, there are the corresponding sets G and B defined by

$$G = \{m|y_m = 1\} \qquad \text{and} \qquad B = \{m|y_m = 0\}.$$

One of the 2^N square bracketed terms in the summation will be

$$A_1 = q^{\#G^*}(1 - q)^{\#B^*}\varepsilon^{\#B^*} \prod_{m \varepsilon G^*} P(x_m|a, b_m, y_m = 1).$$

This term corresponds to the value of $y_1, \ldots, y_N$ reflecting the true but unknown state of the outliers. We will compare this term to an arbitrary different square bracketed term

$$A_2 = q^{\#G}(1 - q)^{\#B}\varepsilon^{\#B} \prod_{m \varepsilon G} P(x_m|a, b_m, y_m = 1).$$

Upon comparing A_1 with A_2, we have

$$\frac{A_1}{A_2} = \frac{q^{\#G^*}(1-q)^{\#B^*}\varepsilon^{\#B^*}\prod_{m\varepsilon G^*} P(x_m|a, b_m, y_m = 1)}{q^{\#G}(1-q)^{\#B}\varepsilon^{\#B}\prod_{m\varepsilon G} P(x_m|a, b_m, y_m = 1)}$$

$$= q^{\#G^* - \#G}(1-q)^{\#B^* - \#B}\varepsilon^{\#B^* - \#B}\frac{\prod_{m\varepsilon G^* - G} P(x_m|a, b_m, y_m = 1)}{\prod_{m\varepsilon G - G^*} P(x_m|a, b_m, y_m = 1)}.$$

But $\#G^* + \#B^* = N$ and $\#G + \#B = N$. Hence,

$$\frac{A_1}{A_2} = \left[\frac{\varepsilon(1-q)}{q}\right]^{\#G - \#G^*}\frac{\prod_{m\varepsilon G^* - G} P(x_m|a, b_m, y_m = 1)}{\prod_{m\varepsilon G - G^*} P(x_m|a, b_m, y_m = 1)}.$$

Each term for which m belongs to $G^* - G$ produces a value of $P(x_m|a, b_m, y_m = 1) \gg 1$. Each term for which m belongs to $G - G^*$ produces a value of $P(x_m|a, b_m, y_m = 1)$ which is small in comparison to $\varepsilon(1-q)/q$. Hence

$$\frac{[\varepsilon(1-q)/q]^{\#G - \#G^*}}{\prod_{m\varepsilon G - G^*} P(x_m|a, b_m, y_m = 1)} = \prod_{m\varepsilon G - G^*}\left[\frac{\varepsilon(1-q)}{qP(x_m|a, b_m, y_m = 1)}\right] \gg 1.$$

Therefore, it follows that $A_1 \gg A_2$.

This argument suggests that under the assumed conditions there will be one term among the 2^N terms in the summation which will be a dominant term. We simplify the estimation of a by estimating a by the value $\hat{a}$ where $\hat{a}$ maximizes the dominant term. That is, for each of the 2^N values of $y_1, \ldots, y_N$, there will be a corresponding value $\hat{a}(y_1, \ldots, y_N)$ which maximizes

$$P(\hat{a})\prod_{n=1}^{N}[P(x_n|\hat{a}, b_n, y_n)P(y_n)].$$

The dominant term corresponds to that value $y_1, \ldots, y_N$ for which

$$P(a(y_1, \ldots, y_N))\prod_{n=1}^{N} P(x_n|a(y_1, \ldots, y_N), b_n, y_n)P(y_n)$$

$$\geq P(a(z_1, \ldots, z_N))\prod_{n=1}^{N} P(x_n|a(z_1, \ldots, z_N), b_n, z_n)P(z_n)$$

for all values of $z_1, \ldots, z_N$.

V.3. Discussion

This procedure is related to and supports the random consensus method of Fischler and Bolles. Fischler and Bolles use a randomly chosen small number of

observations to determine an estimate $\hat{a}$ of a. Instead of substituting this estimate into

$$\prod_n P(x_n|\hat{a}, b_n, y_n = 1)$$

where the product is taken over those n having suitably high values of $P(x_n|\hat{a}, b_n, y_n = 1)$, Fischer and Bolles consider the estimate $\hat{a}$ a reasonable one if the number of terms in such a product is higher than that a produced by any other randomly chosen small number of observations. This step is, in effect, a verification step for the estimate $\hat{a}$.

The estimation technique suggested in Section IV calculates the qualified product instead of just counting the terms which would be in the product. The qualified product of the confirmation is related to

$$\prod_{m \epsilon G^*} P(x_m|a, b_m, y_m = 1)$$

the product expression appearing in the dominant term of the summation in Section IV.

VI. HYPOTHESIS VERIFICATION

The hypothesis verification functions, so essential in expert systems and in the control mechanism of vision systems, can also be posed in a precise way admitting to well-defined optimal solutions. The difficulty in posing the problems is how to do it in a way in which uncertainty can be properly handled.

In this section we suggest one possible approach. Let $R_1, \ldots, R_N$ denote N rules which the system has in its knowledge base (short term and long term). These rules can be any predicate calculus proposition of the if–then form. For our model of hypothesis verification, in any instance in which verification must be done, some rules hold and some rules do not. That is, not all the rules are true all the time; even if the premise of a rule is satisfied for the current instance, there is some chance its consequence may be false. Thus associated with each rule R_n, is a random variable y_n which takes the value binary one if R_n is true and the value binary zero if R_n is false.

In this notation, R_n designates the semantic content of the rules and y_n designates whether or not the rule holds. The probability $P(y_1, \ldots, y_n|R_1, \ldots, R_N)$ is the conditional probability of $y_1, \ldots, y_N$ being the truth state of rules $R_1, \ldots, R_N$ given the semantic content of rules $R_1, \ldots, R_N$. Our first assumption is that the truth state of the rules is independent of their semantic content. This means that

$$P(y_1, \ldots, y_N|R_1, \ldots, R_N) = P(y_1, \ldots, y_N).$$

Our second assumption, which is a hidden assumption in most hypothesis verification techniques, is that the truth state of each rule is independent of the truth state of every other rule. That is,

$$P(y_1, \ldots, y_N) = \prod_{n=1}^{N} P(y_n).$$

It is clear that this is the simplest of all possible assumptions that can be made about the interaction effects of the truth state of one rule or the truth state of another rule. More complex assumptions about this interaction can be made and a more complex procedure for hypothesis verification will result. We, however, use the simplest assumption here for illustration purposes.

Let h designate the hypothesis to be verified. Verification of h amounts to computing the conditional probability that h holds given the knowledge embodied in rules $R_1, \ldots, R_N$. We denote this conditional probability by $P(h|R_1, \ldots, R_N)$. If the computed conditional probability is high enough we say h is verified.

Now we compute $P(h|R_1, \ldots, R_N)$ in terms of a sum of 2^N terms, one term for each truth state of the rule set

$$\begin{aligned} P(h|R_1, \ldots, R_N) &= \sum_{y_1, \ldots, y_N} P(h, y_1, \ldots, y_N, R_1, \ldots, R_N)/P(R_1, \ldots, R_N) \\ &= \sum_{y_1, \ldots, y_N} P(h|y_1, \ldots, y_N, R_1, \ldots, R_N) P(y_1, \ldots, y_N|R_1, \ldots, R_N). \end{aligned}$$

Consider the meaning of the term $P(h|y_1, \ldots, y_N, R_1, \ldots, R_N)$. Of the N y_n's, some will have binary value one and some will have binary value zero. If a y_n has the value binary zero, then R_n is logically false. If a y_n has the value binary one, then R_n is logically true. Let $i_1, \ldots, i_K$ designate the indices of those y_n's which have value binary one and let $j_1, \ldots, j_{N-K}$ designate the indices of those y_n's which have the value binary zero. If the conjunction of $R_{i_1}, \ldots, R_{i_K}, \sim R_{j_1}, \ldots, \sim R_{j_{N-K}}$ semantically imply h, then $P(h|y_1, \ldots, y_N, R_1, \ldots, R_N) = 1$. If the semantic implication does not hold, then $P(h|y_1, \ldots, y_N, R_1, \ldots, R_N) = 0$. We will use the shorthand $y_1, \ldots, y_N \Rightarrow h$ to denote the holding of this semantic implication. Hence the summation for $P(h|R_1, \ldots, R_N)$ simplifies

$$P(h|R_1, \ldots, R_N) = \sum_{\substack{y_1, \ldots, y_N \\ y_1, \ldots, y_N \Rightarrow h}} P(h|y_1, \ldots, y_N, R_1, \ldots, R_N).$$

The above equation conceptually means that we must sum over all truth states of the knowledge base which can semantically imply h. For each such truth state we add to the sum the weight or probability $\prod_{n=1}^{N} P(y_n)$. Computationally we can be a bit more efficient. Those rules, whether they are asserted as true or false, whose premises are not satisfied at any time in the reasoning process do not need to enter into consideration in the summation. The reasoning process itself may use the probability values $P(y_n)$ to determine which rule to try next. Finally, if it only is required to determine if $P(h|R_1, \ldots, R_N)$ is high enough, the summation process need only continue until the threshold value is reached.

VII. CONCLUSION

We have raised the issue of the ad hoc nature of much computer vision research. We have discussed the small amount of replication and comparisons reported. We have tried to illustrate the formalization of the model matching problem and give a suitable motivation for a Bayesian and robust technique for determining the sensor

projection parameters required to do model based matching. The technique is a refinement and generalization to the Fischler Bolles random consensus technique. Suitable experimental comparisons are now needed between it, the Stockman clustering technique, and the Fischler Bolles technique. Finally, we have discussed a Bayesian approach to hypothesis verification, an important element in a high level vision control mechanism.

REFERENCES

1. R. O. Duda and P. Hart, *Pattern Recognition and Scene Analysis*, pp. 436–441, Wiley, New York, 1973.
2. M. Fischler and R. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Comm. ACM* **24**, No. 6, 1981, 281–395.
3. W. Perkins, A model based vision system for industrial parts, *IEEE Trans. Comput.* **C-27**, 1978, 126–143.
4. L. G. Roberts, Machine perception of three dimensional solids, in *Optical and Electro Optical Information Processing* (J. T. Tippett *et al.*, Eds.), pp. 159–197, MIT Press, Cambridge, Mass., 1965.
5. G. Stockman, S. Kopstein, and S. Benett, Matching images to models for registration and object detection via clustering, *IEEE Trans. Pattern Anal. Mach. Intell.* **PAMI-4**, No. 3, 1982, 229–241.
6. R. M. Haralick and G. L. Elliott, Increasing tree search efficiency for constraint satisfaction problems, *Artificial Intelligence* **14**, 1980, 263–313.