Landmark Identification and Pose Determination in Unstructured Environments

Francesco Callari, Gilbert Soucy, and Frank Ferrie

TR-CIM-96-08 Jun 20, 1996, revised November 22, 1996

Centre for Intelligent Machines
McGill University
Montréal, Québec, Canada

Postal Address: 3480 University Street, Montréal, Québec, Canada H3A 2A7
Telephone: (514) 398-6319 Telex: 05 268510 FAX: (514) 398-7348
Email: cim@cim.mcgill.ca

Landmark Identification and Pose Determination in Unstructured Environments

Francesco Callari, Gilbert Soucy, and Frank Ferrie

Abstract

This paper introduces a novel model-based technique to determine the pose of complex, non-cooperatrice 3-D objects from laser rangefinder images. Geometrical models of object parts are probabilistically matched against a model database, and various sources of uncertainty are carried through the process, yielding local, point-by-point estimates of position and predicted error. Promising experimental results are presented and discussed.

Résumé

Ce rapport introduit une nouvelle technique basée sur l'utilisation de modèles pour déterminer la pose d'objets complexes à partir d'image télémétriques. Les modèles géométriques de parties d'objets sont associés statistiquement aux modèles d'une banque de données et plusieurs sources d'incertitudes sont propagées tout au long de ce procédé menant à un estimé local, point par point, de la position et de l'erreur de prédiction. Des résultats expérimentaux prometteurs sont présentés et discutés.

1. Introduction

Model-based vision refers to a class of approaches in which features or structures computed from images are used to index a database of models corresponding to known objects. Approaches can be differentiated by the types of features/structures used, the methodology of indexing, and the nature of the database models (e.g. generic or specific). Although successful in structured environments, e.g. the factory floor, model-based approaches are often difficult to apply in unstructured ones. This is especially true in environments where imaging conditions and/or correspondence between object and model can vary significantly. For these reasons much research has focused on techniques to minimize variation in the process of image formation (e.g. structured light, optical triangulation, time-of-flight) and on the identification of image structures that are invariant or quasi-invariant under projection [50] [57].

The approach described in this report exploits both of these ideas in a novel way to create an object recognition system that is robust with respect to the kinds of imaging conditions found in unstructured environments and that does not rely on specific object models. A laser rangefinding system based on synchronized scanning [69] combined with sophisticated filtering and analysis techniques is used to reliably extract image structures that are largely invariant with respect to viewpoint under realistic conditions of image noise. Parametric models fit to this data are subsequently used to characterize the intrinsic geometry of these structures in terms that are readily identifiable by humans (e.g. spheres, cylinders, rectilinear solids, etc.). Recognition of the particular shapes is accomplished automatically. For example, a laser rangefinder image of an automobile would result in a description consisting of a rectilinear solid supported by 4 cylinders. Information extracted from model parameters would further detail the dimensions of each element along with their spatial relationships and parametric uncertainties.

This approach greatly facilitates both the building of descriptions (the database generation problem) as well as the recognition of objects (computational complexity) and further provides a means of quantifying physical properties and their uncertainties (object pose and shape). To illustrate these points we will consider a concrete example that is of considerable importance in space assembly context and that also has strong commercial implications (component alignment in automated surface mount inspection systems). The specific application is pose determination of structures for the purpose of precision alignment, e.g. docking of space structures or insertion and soldering of complex surface-mount components. In both cases the vision system must be able to identify landmarks in the data (identification and parameterization of local models) and combine these measurements into a global prediction of the position and orientation of the object.

There are two stages to this process. Offline, reference models are created for each object that must be located in the environment. Although one could potentially rely on the system to perform this task automatically, in practice some human interaction is usually required in model (landmark) selection[1]. This task is directly analogous to the current practice of placing cooperative targets on structures according to visibility constraints. As currently

[1] The term *landmark* will be used to indicate the particular usage of a model in the role as a landmark for localization purposes.

implemented in our laboratory, this amounts to a user selecting landmarks that are automatically identified by the system with a pointing device such as a mouse on a graphical display system. The parameters of each landmark model selected and their spatial configuration are subsequently used by the system to recognize and determine the pose of the object dynamically online. We refer to the latter as the *global description*.

Online, information acquired from the laser rangefinding system is processed in identical fashion, resulting in a scene description consisting of a set of parametric models and their spatial configuration. The task performed by the system next is to extract the global description, if it exists, from the scene description. Fortunately the combinatorics of the resulting search are greatly reduced by the structure of the spatial configuration as will be described later on. The search process determines whether a known object is present in the scene. Information about the state of that object is available from the parameters of the global description. For example, in the docking or alignment processes it is important not only to determine how the object is aligned in space, but also to quantify the estimated accuracy of this measurement. This will inform the system whether it is safe to proceed if the measurement is within tolerance, otherwise to gather further measurements until the measurement is within spec.

This report, a result of work done in collaboration with Hymarc Ltd., presents a proof-of-concept of our system and demonstrates its performance on a real-world task that is typical of both the space assembly and commercial environments. We begin in Section 2, with a survey of the literature related to the major components of our system, reporting on the most relevant papers in detail. The algorithm devised for performing this task will then be presented in detail in Section 3, followed by results on real range data acquired from a complex object that typifies the task in Section 4. Finally, we conclude with some discussion and a pointer to future work proposed for Phase II of this project in Section 5.

2. Literature Review

In this section, we present a review of the state of the art in fields related to this project.

- Range image segmentation.
- View correspondence.
- Model-based object recognition.

Range image segmentation is the process by which subsets of data corresponding to relevant object parts are identified in the measured data.

When the uncertainty associated with a particular part localization becomes too large, it will be necessary to continue the experiment with data acquired from a different vantage point. In these situations, view correspondence will be required to integrate the information from all the viewpoints in order to refine the solution.

Object recognition will be the formal way to identify with a given level of confidence the parts, which can then be used to locate an object in space with respect to the sensor.

Various automatically indexed databases have been searched for relevant literature for this survey, including the McGill Peruse system and various university/industrial archives available on the Internet. A list of various papers has been compiled, and the most relevant of these will be briefly reviewed in the following sections.

2.1. **Range Image Segmentation.** A key problem in the interpretation of range data is the partitioning of a shape into its principal components or parts. Often referred to as range image segmentation, this problem consists of assigning consistent labels to the pixels of an image according to some metric or physical property.

In light of the latter definition, the problem might look simple but it is in fact very complex. The complexity of the problem comes from its fuzzy and incomplete definition. The measure of success of any segmentation algorithm is most often subjective, based on some human notion of what is a part, which involves a global understanding of the scene. For some particular objects, different human observers will disagree on the segmentation making the task of automating such a process very difficult.

Over the years, researchers have tried to come up with a consistent theory of shape and part decomposition. Kimia [42], for instance, has studied the evolution of 1-D closed shapes (object contour) through two types of deformation similar in nature to morphological operators: diffusion and reaction. Diffusion applies a smoothing proportional to the contour curvature while reaction tends to make apparent the part boundaries by applying a constant deformation to the contour. Kimia argues that the segmentation results obtained with his method are similar to what a human operator would outline.

While not directly applied to range image, Kimia's theory has paved the way to a growing interest in the so-called scale space and its potential beneficial effects. The scale space is a multi-level representation of a shape through successive application of a smoothing operator. The desired effect sought is the elimination of noise while preserving the interesting features assuming that these features exist only within a certain range of scale.

Chelberg and Hi [18], for example, have developed a segmentation based implicitly on the scale space by making use of regularization techniques. They proposed a new functional that varies the amount of smoothing according to the gradient of the data. An iterative application of reconstruction using this new functional improves the signal/noise ratio of the noisy input image with good preservation of discontinuities. The results indicate that the algorithm performs especially well on noisy range images.

Traditionally, Gaussian smoothing has been chosen for reason of simplicity to perform scale space mapping. Unfortunately, this operator is known to affect considerably the precise location of the features to be preserved [49]. This fact has resulted in complex algorithms to perform feature tracking through scale space [88]. Lejeune and Ferrie [49] have introduced a non-linear diffusion operator whose primary strength resides precisely in its ability to preserve feature location through scale. By construction, this operator is guaranteed to keep untouched regions of interest while quickly filtering out sensor noise and more progressively the surface texture.

The work of Lejeune and Ferrie is centered around a definition of an object as a solid mass composed of convex parts. According to this definition, inspired by the work of Hoffman and Richards [68], the boundaries between parts are closed concave valleys. The non-linear diffusion is applied to express the data in the right range of scale and leads to a hybrid representation of the boundaries: edge based and region based. The two aspects of this representation are considered as complementary information and are integrated by a relaxation process to produce the final output of the system. The method has proved to produce stable results even in presence of severe noise.

Yokoya and Levine [89] also considered a hybrid description of the part boundaries. In this case, the initial region map is extracted from differential properties of the surface (mean and Gaussian curvature map). The intrinsic noisy characteristics of such a map [81] are compensated by combination of the information from the roof and jump edge maps. The article makes no mention of any pre-conditioning of the data and therefore makes no use of the scale space. Ghosal and al. [29] present a similar hybrid approach where edge map and region map are integrated into a unique and consistent result.

Top-down approaches, or the use of a priori knowledge of a specific problem at hand, have lead to numerous segmentation algorithms. For instance, Gupta and Bajcsy [32] [33] have proposed a geometric model-driven framework for segmenting dense range data of complex 3-D objects into their constituent parts. Their scheme employs a local-to-global iterative regression approach of searching for the best piecewise bi-quadric description of the data. Superquadric models are recovered by a global-to-local residual-driven procedure.

Other methods have tried to segment the range data into planar or quadric regions. Kaveti, Khwang and Wang [41] have proposed an algorithm where a combination of split-and-merge and region growing algorithms is performed. Planar regions are extracted first by a clustering approach, wherein, during each iteration the coordinate along which the separation of regions is highest is selected. Taylor, Savini and Reeves [79] presented a technique for rapidly dividing surfaces in range imagery into regions satisfying a common homogeneity criterion. The result is a segmentation of the range information into approximately planar surface regions.

The use of other complementary sources of information such as color or intensity can help in simplifying the problem. For instance, Seetharaman and Chu [71] considered the segmentation of a range/intensity image pair of a 3-D scene for model based 3-D object recognition. They noted that, in some cases, at the intermediate levels of abstraction (such as local-shape from shading) the uncertainties that exist in range and intensity images are of complementary nature.

Lately, a group of researchers [35] have tried to put together an objective way of comparing the results of various segmentation algorithms. They made available a set of 40 range images with corresponding ground-truth (manual) segmentation. The evaluation is performed of the basis of five region classifications: correct detection, over-segmentation, under-segmentation, missed and noise. They defined an interesting set metrics to automatically compare the results with ground-truth.

In general, however, the evaluation of any range image segmentation algorithm must be done in consideration of the class of problem the method was developed for. A specific algorithm might well return good or bad results when the nature of the sensor noise changes, under different scales, when the class of object is changed or when the task is changed. Even ground-truth segmentation is subjective with respect to some criteria or task. Range image segmentation covers a very wide field of possible applications and require, for this reason, a minimum of contextual and a priori information in its design and its evaluation.

2.2. **View Correspondence.** Building a complete 3-D model of any object necessarily implies the acquisition of data from different vantage points. The data from all these viewing positions is expressed in a camera-centered frame and needs to be remapped into a global coordinate frame. View correspondence is the solution to this local to global problem: what is the *set of motion parameters* that will bring in alignment the data from all these views? It is in fact very similar to a jigsaw puzzle (in 3-D), the pieces being the different views.

With regard to this problem, two approaches can be taken:

- Use a very precise manipulator and apply careful calibration.
- Use a relatively imprecise manipulator, calibrate as well as possible and fix the remaining errors by software.

In general, the choice made is mainly a question of expense. Manipulators precise up to the level required to reverse engineer a large set of objects (CMM - coordinate measuring machines) cost in the order of hundreds of thousands of dollars. The more precise they are, the more expensive they are and the relationship is not linear.

It gets even worse when the level of precision approaches the micron. The manipulator needs to be used in a temperature and humidity controlled environment, adding drastically to the cost. While being expensive in general, this solution provides a quick and reliable solution to the view correspondence problem (as reliable as your hardware).

Expense is not always the only factor. Sometimes, it is simply impossible or impractical to use such a precise manipulator. In some space applications for instance, only an estimate of the motion between two consecutive views can be obtained via inertial sensors. In such a case, the data has to be matched across views to solve correspondence.

The computer vision community has given a lot of attention to this problem over the last few years as laser range finders have become more common. The level of sophistication of the different approaches varies largely. It ranges from a simple minimization of the distance between the data points across views to the evaluation of higher level view-invariants (such as surface curvature) that can be matched more accurately across views.

For instance, Chen and Medioni [19] have developed a method that can be used to refine an estimate of motion by minimizing the pixel distance across views. Bergevin, Laurendeau and Poussart [5] have tried to eliminate the need for an estimate by correlating the hierarchical triangulations of each view. Soucy and Ferrie [75] introduced a method to cope with non-rigid body motion by calculating a dense grid of motion parameters covering the whole surface.

While the software approach to the view correspondence problem might seem attractive, it has a major drawback. It is not guaranteed to always give the right answer: its precision depends on the data itself (e.g. it is not possible to register two views of a plane). The data should be somewhat *unique* locally so that the match can be done unambiguously. Unfortunately, it is, in general, impossible to control this factor and this will cause even the most sophisticated algorithms to fail.

2.3. **Model-Based Object Recognition from Range Imagery.** Very comprehensive, though rather dated, surveys on techniques for Model-Based Object Recognition (MBOR) can be found in the works by Besl and Jain [6], and Chin and Dyer [20]. A more recent one is in Arman and Aggarwal [2], which focuses on range imagery. In the following we shall often refer to the last two works, integrating with other sources where necessary.

In a number of applications, among the goals of a vision system is to identify and locate a specified objects in a scene. This is often achieved by using *models* of the objects, which in most cases contain information about their geometry, though other physical properties (e.g. color, mass, thermal or stress properties) may be modeled as well. A vision system which makes use of object models is called *model-based*, and the general problem of identifying objects according to their models, and determining their exact location and orientation in a scene is referred to as *model-based object recognition*.

An ideal MBOR system should be able to locate and identify objects in a scene regardless of their shapes, viewing direction, partial mutual and/or auto occlusion. Robots with such vision capabilities may carry out tasks such as the handling of mechanical parts with fewer specifications than are currently required, and with more tolerance of minor disturbances.

The design an MBOR system involves the specification of:

1. Sensor requirements for data collection.
2. Methods for data pre-processing (filtering, smoothing, etc.).
3. Methods for figure-ground separation and scene segmentation.
4. Used models (representations).
5. Methods for recovering the models from the data, for regularizing the models and for evaluating the fit accuracy.
6. Methods for matching the models fitted to the data to a known database of object models, and for evaluating match ambiguity.
7. Methods for building a database of object models.

This section will focus in particular on the modelization aspects of the framework (the last four points).

2.3.1. *Choice of Models.* The choice of models of a particular mathematical class formally expresses prior knowledge about the modeled object (presence or absence of symmetries, curvature scales, degree of shape smoothness, etc.).

Moreover, the choice of a model class strongly influences the overall architecture of an MBOR system, since (a) it fixes the goal of the model fitting methods, usually by determining the form of a cost functional to be optimized by the fitting procedure (parametric regression on fixed form), and (b) determines the content of the object model database used for matching.

The feasibility of a model class for recognition is dictated by requirements of (1) absence of ambiguity in the database (no two objects among those to be recognized may be represented by the same model) (2) uniqueness (an object may have only one model, or a well known set of possible models) (3) graceful sensitivity to missing data and noise, (4) convenience to manipulate and store.

Modeling schemes can be broadly classified into three main classes [2]:

- Surface-based.
- Discontinuity-based.
- Volumetric-based

Surface-based

Surface-based schemes use object surface properties, such as surface normal and Gaussian curvature, to approximatively and collectively describe the measured data points.

Gaussian Sphere. In particular, in schemes based on the Gaussian sphere concept, surface normals are estimated in correspondence to the data points, or on a regular grid superimposed on them, and each normal is mapped onto the corresponding point of the unit sphere [23] [37]. The mapping is strictly local at elliptic and hyperbolic points only, allowing for the unambiguous representation of globally convex objects only. The representation is invariant w.r.t. rotation, but information about object absolute position and overall size is not preserved. However, several modifications to this basic scheme have been proposed in the literature to overcome these limitations.

Horn's extended Gaussian sphere [36]. Each point on the sphere is assigned a weight equivalent to the corresponding surface patch area. Kang and Ikeuchi [40] associate with each surface normal a complex number whose magnitude represents the patch area, and whose phase represents the patch distance from the origin of a coordinate system - this information being essential for recovering object position and pose. Multiresolution representations via tessellation of the sphere have been proposed by Korn and Dyer [45]

All Gaussian sphere representation schemes share a common problem: in complex scenes with several objects, proper object partitioning must be achieved prior to modeling to avoid improper interaction among the representations of each object on the sphere.

Discontinuity-based

Discontinuity-based models focus on representing information about the locations where surface characteristics change. Sets of such points constitute curves in 3-D space, often approximated by polynomial curves or rational B-splines, which can be efficiently stored and manipulated. Surface information, such as concavity/convexity, is lost in these models.

Surface primal sketch. This model class has been proposed by Ponce and Brady [64]. It is based on the detection and representation of several types of discontinuities: *steps*, where the depth map has jump discontinuities, *roofs*, where the surface normal is discontinuous, *smooth joints*, where the principal curvature is discontinuous, and *shoulders*, where two roofs combine. Detection of the four types of discontinuities is performed after smoothing with a Gaussian kernel at multiple scales and estimating on each point the principal directions and curvatures, and the zero-crossings of the Gaussian curvature.

Aspect graph. First proposed by Koenderink and van Doorn [43] [44], aspect graphs are graphs whose nodes represent distinct viewpoints of a 3-D object, and whose arcs represent the transformations of *visual events*. The set of all viewing directions (mapped on a Gaussian sphere) is partitioned in "stable" regions, where small changes in the viewpoint do not cause significantly the aspect of an object. The transitions between stable regions individuate *accidental views* where the aspect changes rapidly. The first applications considered the simple case of polyhedral objects [61], but more recently applications to curved surfaces have been proposed: generalized cylinders have been considered by Eggert and Bowyer [25] and Kriegman and Ponce [46]. The major disadvantages of these types of representations are the high construction time, excessive storage requirements for each model, and limited class of objects for which workable solutions are available. An extensive review of these methods can be found in [10].

Volumetric representations

This class of representations comprises models of the volume occupied by objects rather than their contouring surfaces. Most of these schemes are efficient at describing objects of simple shape ("parts", or articulated models composed of simple parts), and such that symmetries can be exploited.

Octrees. [38] are based on volume decomposition into a tree-structured hierarchy of cubes. Variations on the theme include the use of wedges and parallelepipeds and as basic elements [58], or storing in the tree leaves information on the corresponding body plane, edge or vertex [47]. These models are mostly used for shape modeling, rather than recognition, due to their sensitive dependence on the viewpoint.

Superquadrics. Known since the XVI century, this class of solid shapes has been introduced in computer graphics by Barr [3], although Pentland's work [59] [60] largely contributed to their diffusion in the computer vision community. Superquadrics are solid shapes defined by a spherical product operation on two orthogonal plane curves, and can be regarded as derived from the basic quadric surfaces by considering fractional exponents of the form $2/\epsilon$ in the quadric canonical form. For example, the parametric equations of the canonical superellipsoid in the first octant are:[2]

$$\begin{aligned} x(\eta,\omega) &= a\,|\cos\eta|^{2/\epsilon_1}\,|\cos\omega|^{2/\epsilon_2}; \\ y(\eta,\omega) &= b\,|\cos\eta|^{2/\epsilon_1}\,|\sin\omega|^{2/\epsilon_2}; \\ z(\eta,\omega) &= c\,|\sin\eta|^{2/\epsilon_1}; \\ & \quad a, b, c, \epsilon_1, \epsilon_2 > 0; \end{aligned}$$

[2]The surface is closed and symmetric w.r.t its center and principal planes, hence a specification in an octant suffices to fully determine the shape.

where η and ω are latitude and longitude respectively, and a, b, and c are the semi-axis lengths along x, y and z respectively. The equations become identical to those of an ordinary ellipsoid when both shape parameters ϵ_1 and ϵ_2 equal 1, while a wide variety of shapes can be obtained by changing their value: box-like shapes are obtained for $\epsilon_1, \epsilon_2 \simeq 0$, cylinders for $\epsilon_1 \simeq 0$, $\epsilon_2 \simeq 1$, diamond-like for $\epsilon_1, \epsilon_2 \simeq 2$. Globally convex shapes are obtained for $|\epsilon_1|, |\epsilon_2| < 2$, which is the usual case in modeling. A superellipsoid in generic position and pose is specified by 11 parameters: the above $a, b, c, \epsilon_1, \epsilon_2$, the three coordinates of the center and the three Euler angles of its principal frame w.r.t. the world coordinate system.

Boult [9], Bajcsy and Solina [74] and Ferrie *et al* [27] present frameworks for optimally fitting superellipsoids to depth maps. Also with reference to superellipsoids, Whaite and Ferrie [82] [83] consider the case of optimal active model refinement, and Callari and Ferrie [14] [13] the case of recognition-oriented optimal active data collection.

Constructive Solid Geometry. Complex shapes are represented by regularized boolean operations on primitive solids, such as blocks, spheres, wedges, cylinders, superellipsoids. Objects are represented by trees whose leaves are primitives, whose inner nodes are boolean combinations (union, intersection, difference, complement), and whose arcs individuate the operands of each operation (primitives or results of previous operations). A regularization procedure must be applied to ensure that the result of all operations is a solid (e.g. the intersection of two touching objects is not a solid). This kind of modelization scheme is mostly useful for model construction (as in CAD/CAM systems), but presents drawbacks as far as recognition is concerned: the object representation is often not unique, and may not model accurately finely sculptured objects with a reasonable number of primitives.

Generalized cylinders [7] model shape by sweeping along a path in 3-D space (the "spine") a 2-D curve referred as "cross section". Generalized cones are obtained if an omotetic scaling is applied to the swept curve along the spine. Ponces's [65] "straight homogeneous generalized cylinders" have a straight spine, but allow for arbitrary cross section and sweeping rule. "Curved solids of revolution" allow for non constant spine, cross section, cross section scaling and sweeping rule.

2.3.2. *Model Recovery.* The model recovery problem often turns out to be mathematically ill-posed in the sense of Hadamard [62], i.e. with no exact and unique solution and/or with highly sensitive dependence of the solutions on the data (and hence on the noise affecting the data). To correct this problem the fitting procedure is often regularized by appropriate functionals. These essentially bias the convergence of the fitting method toward those models within the chosen class which have high prior probability to be correct [51].

Several of the above representations can be computed directly on the data, and no explicit fitting/optimization process is required. In this case the regularization process essentially addresses the stability issue, and often is reduced to data smoothing.

This is not so for nonlinear models like superquadrics, whose recovery involves iterative optimization procedures of appropriate cost functionals. However, models of this class are not subject to combinatorial explosion problems, and are amenable to efficient optimization algorithms (Levenberg-Marquardt, variable metric, conjugate gradient [63] [66] [67]. Moreover, computational costs become been less and less of an issue with the consistent increase in speed - and decrease in cost - of computing machinery.

The outcome of a model fitting procedure is a particular set of models that optimally fit the segmented parts of the scene, given the chosen model class and (usually) in the least squares sense. Because of sensor noise and model inaccuracy the optimal models themselves will be generally affected by uncertainty. This can be expressed as an *a posteriori* (that is, after receiving the data) probability distribution over the models, with the Bayesian interpretation of probability as a measure confidence [11] [73] [78]. For the least squares criterion, a Gaussian approximation to the posterior distribution is often used, locally in the neighborhood of the optimal model, with covariance matrix equal to the inverse of the Hessian of the fitting cost function at the optimum[3]. The relative sizes of the eigenvalues of the covariance matrix then define the "well determined parameters" of the model, i.e. the model parameters that are well determined by the data, with the others mainly set by the prior instead [31]. This quantification of model uncertainty can be effectively exploited along two lines:

- For active vision systems, to determine which new sensor positions are most useful to refine the model by further data collection (optimal experiments [26], active data selection [53], autonomous exploration [86]).
- For the purposes of model selection (Bayesian "Ockam Razor"), to score alternative models according to both their goodness-of-fit to the data and their relative complexity [31] [51] [73] [77].

2.3.3. *Model Matching.* The next logical step in MBOR is matching the models recovered from the data with those stored in a previously built database. Matching means establishing, ideally, a one-to-one correspondence between the models recovered from the data and "nominal" models built on database objects.

As indicated above, the actual form of the matching is strongly dependent on the particular class of models used, although some general considerations are still possible, concerning goal specifications and accuracy scoring.

With absolute generality, the matching process itself can be regarded as pattern recognition, and cast in the framework of classic statistical decision theory. The objective of the matching is then formulated as one of deciding on a pairing between the recovered model and one database item, based on the posterior probability of each database object given the data, and such that the statistical risk involved in a misclassification or rejection is minimized. Deterministic schemes can be regarded as particular cases with degenerate probability distributions. Theoretical performance limits and criteria can then be stated in a common language (essentially Bayes theorem and its consequences), and approximate performance figures can be assessed for practical purposes [24] [70].

Tree search

This is a classic deterministic matching technique(the Interpretation Tree made famous by Grimson and Lozano-Perez [30] goes back to 1987), sequential and hierarchical in character. A discrete set of recovered model features is determined, and they are used in turn to score the set of match candidates in the database w.r.t. consistency with the detected feature, thereby building a tree whose node are labelled by $(f_i,\, m_j,\, s_{ij})$

[3]The (regularized) cost function is assumed at least twice differentiable around the optimum model, and dominated by the quadratic term of its Taylor expansion.

triplets, respectively indicating a model feature, a candidate model and a degree of agreement (a binary value, in the simplest case). The idea is to look for a path in the tree such that there exists a consistent rigid transformation from a database object to the recovered model. This is obviously subject to combinatorial explosion, and hence most proposed methods concentrate on techniques for pruning the search tree, thereby pruning the search space.

A work representative of this school of thought is Flynn and Jain [28]. A match is represented by a triplet (M,A,T), where M is a candidate object model, A is a path in the search tree and T a rigid transformation. The features considered are cylindrical, spherical and planar surface patches, and the tree is constructed using an ordering of sensory data. The tree is pruned by applying a series of unary constraints (surface area, type and various attributes of each feature), followed by a series of binary constraints (rotation estimates using pairs of features, mutual orientation of pairs of features, visibility and distance of parallel planes. The surviving paths are verified against synthetic images generated from the model database via a CAD system.

Tree search is a conservative approach since it considers all possibilities. This is both a strength and a weakness: it is robust at the expense of efficiency [87].

Graph matching

A different approach consists of determining, both from the recovered and the database models, graphs whose nodes represent surface patches or volumetric parts, and whose arcs represent neighborhood relations among them. The problem of matching can then be treated as one of determining a graph isomorphism, i.e. a topology-preserving transform that brings the recovered graph onto a database's one. Similar to the above tree search approach, several levels of pruning are usually required to reduce the space of possible matches.

Among the major works in this area, Fan [22] represents objects and models by attributed graphs, where the attributes represent adjacency types (convex, concave, jump and limb), curvature, orientation and distance among centers of inertia. After an initial pruning, based on number of nodes, number of planar nodes and visible 3-D area, the match is performed on the basis of the largest number of matching attributes. Jain and Hoffmann [39] use an evidence-based method, in which recovered features are treated as conditions, from which evidence rules are derived. Each rule consists of the features it applies to, the bounds on their values, and the number of occurrences of each feature in the database and in the recovered model. A similarity measure is defined as the difference between the measure of support for each match hypothesis and the measure of each refutation according to the rules. Although experimentally successful according to the published conditions, the method has the disadvantage that all evidence rules must be recalculated every time a new object is added to the database. Moreover, experiments with variable viewing conditions are not reported.

Statistical methods

Various recent works in the literature approach explicitly the statistical characterization of pose estimation via a match process.

Already in 1986 Bolle and Cooper [8] proposed a framework for the ML estimation of the location and orientation of 3-D objects from range data, along with methods for the Bayesian estimation and recognition for large data sets. Their approach leverages on some asymptotic approximations for the likelihood of large data sets, whose application is legitimate in the case of large data samples like images. Only simulation results were presented at the time.

Further theoretical grounding was provided in 1989 by Haralick *et al.*'s [34] work on pose estimation from corresponding sets of data (registration problem): four fundamental cases of pose estimation were identified and solved, and the robustness of the solutions with varying levels of sensor noise and types of errors was discussed.

More recently, Camps, Haralick and Shapiro [16] use probabilistic evidence to cut-off or prune a search tree.

Wheeler and Ikeuchi [87] have proposed an object recognition and pose estimation paradigm for range imagery, based on sensor modeling and probabilistic hypothesis generation. Although presenting a thorough probabilistic analysis of the method, the judgement of the final pose estimation results is left to a subjective criterion of acceptability, and no propagation of the pose uncertainty to point predictions is made.

A recent work by Bedekar and Haralick [4], although in a different context (the view correspondence problem) proposes a MAP Bayesian approach which is similar in inspiration to the present one.

Shimshoni and Ponce's [72] work on 3-D object recognition does actually consider the effect on pose estimation of the uncertainties in recovered feature frames, yet they present results on very simple scenes only, and use computer-intensive techniques (Monte Carlo simulation) to approximate the marginal distributions.

Madsen [56], in studying the effect of viewpoint variation on pose estimation, considers the empirical and predictive variances of the pose parameters, but again fails to address their effects on point estimation.

Arbel and Ferrie [1] use statistical estimation model densities for volumetric models, in a recognition by part approach.

Also authors from the "neural network" community have been active in this field.

Chandrasekaran *et al.* [17] make use of a Kohonen self-organizing map to formulate class label prediction algorithms based on evidential reasoning methods and Bayes theorem.

Rainey *et al.* present a statistical multiple object detection and location system base on neural networks, which is currently applied to multi-look evidence accrual for tactical and strategic reconnaissance.

Callari and Ferrie [13] propose a method for actively searching the most discriminative views by correlating the sensitivity of a model to its parameter with a match probability estimated via a connectionist Bayesian classifier.

3. Recognition by Parts and Pose Estimation

We now present an application of model-based object recognition to the problem of pose estimation of a known object in an unstructured environment. The advantage of using recognition in this context is the elimination of special-purpose (or cooperative) landmarks often required for this type of application.

Given a controlled and calibrated environment, solutions involving intensity images and cooperative landmarks have proved to provide reliable answers in most situations. However, when the lighting conditions can vary largely or when no special purpose landmarks are available this type of approach is prone to error.

The problem becomes even more acute when the reachability is limited and/or targets are occluded and must be extrapolated from some visible landmarks. In such cases, precise information such as that which can be extracted from a dense range image can provide considerable assistance.

Our solution rests on the identification of a unique group of landmarks that can be extracted automatically from range data and does not require any modification or addition to the object (e.g. cooperative targets). The identification of this group of landmarks provides an estimate of the object's position and orientation, and can be used to extrapolate the location of arbitrary occluded targets, providing bounds on the accuracy of the result.

The rest of this Section introduces the major steps of our approach.

3.1. **Data Acquisition.** This step involves two components: viewpoint determination and the actual scanning of the object. For some specific applications, the number of points where the object can be scanned might be limited because of lack of reachability of the manipulator or some other physical constraint imposed by the environment. In this context, given an estimate of the object location, the best viewpoint among all the possible ones is chosen based on factors such as the scanner-object distance and the orientation of the surface with respect to the scanner. The implementation details of this selection are left the second phase of the project.

The actual scanning of the object is in general a device-specific task. For our purpose, we consider the range scanner as a black box providing dense range data of a known characteristic error.

3.2. **Range Image Segmentation.** The range images obtained are segmented using the method developed by Lejeune and Ferrie [49]. This strategy makes use of two complementary representations for the surfaces: one that describes local structures in terms of differential properties (e.g. lines, edges, contours) and the other that represents the surface as a collection of smooth patches at different scales. By enforcing a consistent interpretation between these two representations, it is possible to derive a partitioning algorithm that is both efficient and robust.

A principal characteristic of the approach resides in its scale-space feature where the input range images are gradually smoothed by successive application of a non-linear iterative operator. Through this operator, the surface noise and surface texture are rapidly eliminated while the desired features (surface boundaries) are preserved at large scales at their exact original location. Figure 1 shows the advantage of this smoothing operator over Gaussian smoothing. The over-all process flow of the algorithm is shown in Figure 2.

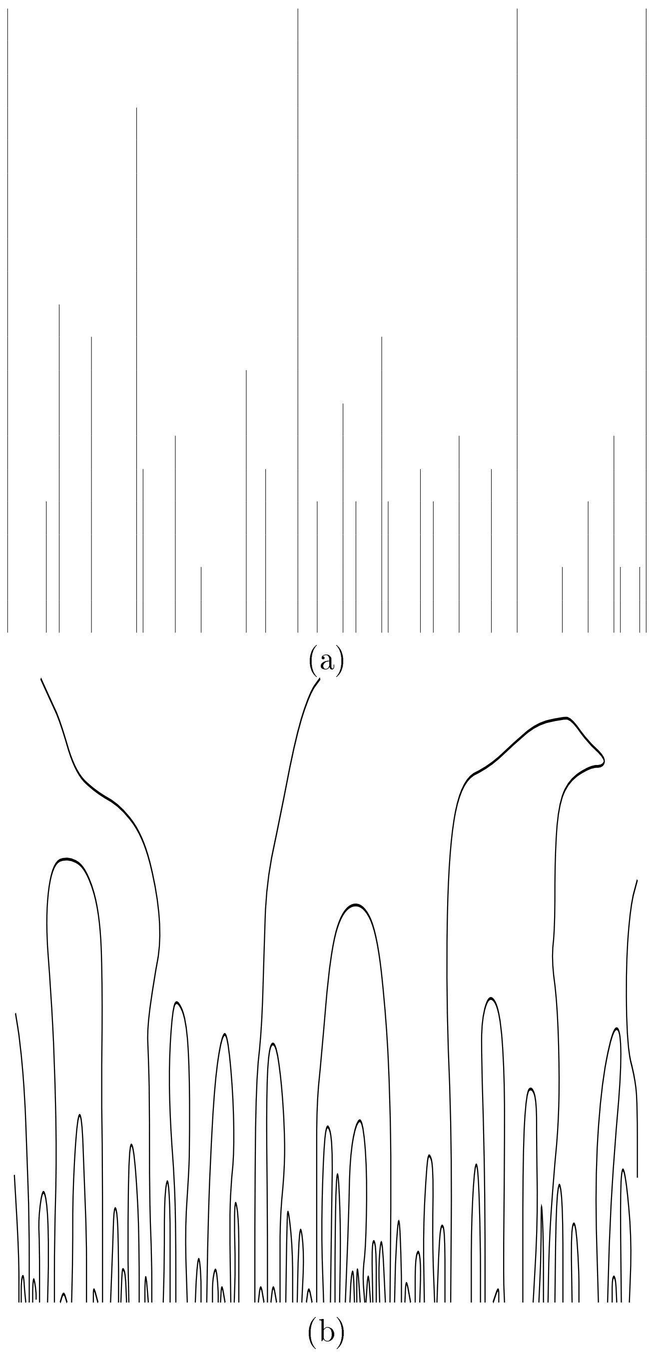
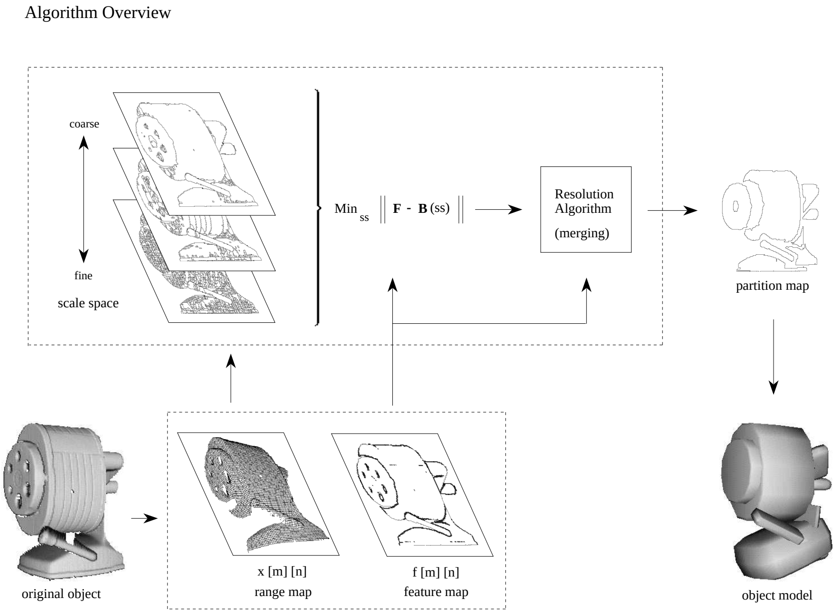


FIGURE 1. Representation of features location through scales for (a) Lejeune and Ferrie's operator and (b) zeros of the second derivative of a Gaussian diffused signal. The features in (a) keep their original location until their disappear while they migrate sometimes largely in (b) and can even merge.



An overview of the parts decomposition algorithm. Given a range map, $x[m][n]$, corresponding to a single view of a three-dimensional object and a feature map, $f[m][n]$, that specifies which elements of $x[m][n]$correspond to putative points on the part boundaries, the algorithm computes a partition map as shown (top right) which makes explicit the set of closed regions in $x[m][n]$that correspond to each part of the object. A scale space of elliptical surface patches (top left) is constructed which is used to approximate $x[m][n]$at different levels of detail. Each patch is guaranteed (by construction) to be smooth, i.e. to be void of C^0, C^1 discontinuities in S and negative local minima in the normal curvature of S. Hence any boundary path is constrained to lie in that subset of S defined by the union of all patch boundaries. The problem of interpolating the boundary can thus be formulated as one of searching the space of patches to find the "optimal" set of closed paths, where "optimality" is defined in terms of how well they support $f[m][n]$. A resolution algorithm is used to determine this optimal set.

FIGURE 2. Process flow: from range data to parts decomposition.

3.3. **Model Fitting.** The segmented parts are approximated with single superellipsoids. Details of the approximation method can be found in [27]. In brief, each segmented part is initially approximated with an ordinary ellipsoids by determining its center of mass and it principal axes of inertia. This is used to start a nonlinear optimization procedure, with cost function equal the sum of the squared radial distances of the data points from the surface of a superellipsoid. The cost function is optimized by the Levenberg-Marquardt algorithm, yielding a (local) optimal estimate for the 11 parameters of superellipsoid fitting the data. As usual in least-squares fitting, the uncertainty about the optimal fit is quantified by a covariance matrix $\mathbf{C}$, obtained by inverting the Hessian of the fit error function at the optimum $\mathbf{m}$.

Note that this way of quantifying the uncertainty requires only a *local* Gaussian approximation of the posterior distribution of the model around the optimum. Although the overall posterior hardly ever follows a normal, particularly in the case of nonlinear models, it is generally dominated by a Gaussian peak in a neighborhood of its maximum, on account of both the central limit theorem and the quadratic behaviour of a non-degenerate error function at the optimum.

3.4. **Part Recognition.** The segmented parts are recognized through their models, by a classifier process producing a conditional probability distribution over candidate objects given the fitted models and the data.

For the problems at hand a data-driven approach to classifier design seems mandatory, although simple parametric methods (e.g. Gaussian approximations "in the large") are likely to fail.[4] However, such a probabilistic recognition framework can be cast in terms of powerful connectionist learning and classification algorithms in a rigorous and structured manner [12] [54] [80], leading to improved classifier performance. The classifier used for the experiments reported here has been implemented with a connectionist architecture, trained and selected using the Bayesian methods for classification networks proposed by MacKay [55] (our experimental results give indications of the adequacy this choice to the task at hand). However, the proposed formalism can be applied to the much wider family of discriminative supervised classification methods

A discriminative classifier is represented by some function $\mathbf{f}(\cdot)$ of the model parameters, providing an estimate of the probability $P(O_i|\mathbf{m})$ that a model parameter vector $\mathbf{m}$ belongs to object class O_i:

$$P(O_i|\mathbf{m}) \simeq f_i(\mathbf{m};\mathbf{w}) \;, \tag{1}$$

where $f_i(\cdot)$ is the i-th component of $\mathbf{f}(\cdot)$, and $\mathbf{w}$ is the vector of the internal parameters of the classifier, to be tuned to a particular data set by training. It is convenient to express $\mathbf{f}(\cdot)$ in the form,

$$f_i(\mathbf{m};\mathbf{w}) = f_i(\mathbf{a}(\mathbf{m};\mathbf{w})), \tag{2}$$

which embodies the internal structure of the classifier in the function $\mathbf{a}(\cdot)$, whose values are translated into probabilities by $\mathbf{f}(\cdot)$. Two forms are commonly used for $\mathbf{f}(\cdot)$: the "logistic" function $f(a) \triangleq e^a/(1+e^a)$ for the binary classification case,[5] and the "generalized logistic" or "softmax" function $f_i(\mathbf{a}) \triangleq e^{a_i}/\sum_j e^{a_j}$ for the multiple class case. The appropriateness of these functions for the output of a classifier is well known to statisticians: they are *the* functions that convert probability ratios into probabilities.

Our approach to discriminative classifier design is based on a particular class of connectionist architectures, the full connected multi-layer perceptrons with one hidden layer, see Fig. 3. The multi-layer perceptrons are among the most useful neural networks for classification problems, since their capability to approximate boundary surfaces of arbitrary complexity makes them universal classifiers.

[4] In the case of superellipsoid models, for example, the inadequacy is due to view-dependent degeneracies [74], that cause the posterior model distribution to variably be uni- or multi-modal, depending on the object and the chosen views.

[5] Both $\mathbf{f}(\cdot)$ and $\mathbf{a}(\cdot)$ become scalar functions in the binary case: a classifier with only one output is used, whose value is interpreted as an estimate of the conditional probability of one class, or one minus the probability for the other.

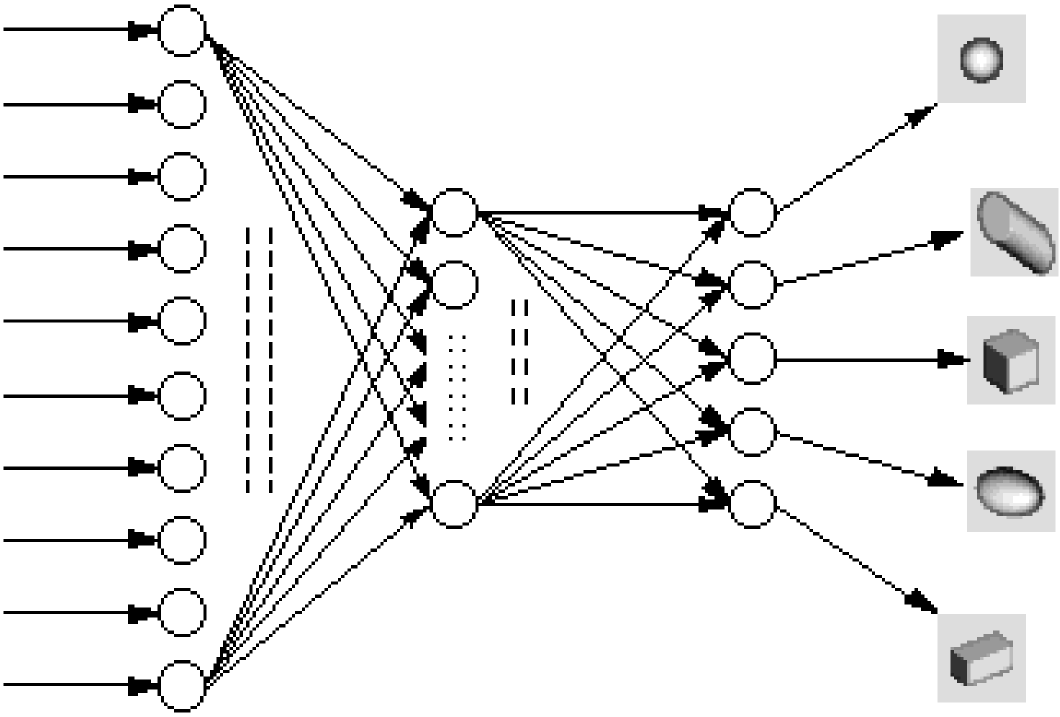


FIGURE 3. A generic representation of the two-layer (one hidden) neural networks we used in this work: each net has as 11 inputs for the shape descriptors and one output for each of the available objects.

An MLP is specified by a pair $\mathcal{H} = (h, \mathbf{w})$, with h being the number of of their hidden units. We use a simple one-to-one correspondence between shape parameters and network inputs, and between object prototypes and network outputs as well, hence all the networks in this work have 11 inputs and a number of outputs equal to the number q of different possible objects (the cardinality of the object database).). The tunable classifier parameters $\mathbf{w}$ then consist the connection weights and unit thresholds of the connectionist architecture.

The parameters are optimized by means of a training set $\{D\}$, consisting of N labelled data, each datum being the pair $(\mathbf{m^{(k)}}, o^{(k)})$ of a model parameter vector $\mathbf{m^{(k)}}$ along with its corresponding object label $o^{(k)}$.

Training

The training procedure and the structure selection technique we use closely follow the hierarchical Bayesian methods proposed by MacKay in his seminal works of 1991–92 [53] [54] [55]. Among the leading ideas of MacKay's approach is that a supervised classifier architecture should be regularized and selected by optimizing its "evidence" as supported by the available data, i.e. its likelihood marginalized with respect to the probability distribution of $\mathbf{w}$.

Specifically, the training of the $\mathbf{w}$ parameters for given a given MLP architecture is accomplished by optimization of a misfit function M, proportional to the log posterior probability of $\mathbf{w}$ given the data. The misfit is expressed as a linear combination of two terms:

- The log likelihood $E_D(\mathbf{w}) = \log P(\{D\}|\mathbf{w}, \mathcal{H}_i)$ of the data given the parameters and the architecture. This term measures how well an architecture and a certain parameter vector "explain" the available data. In our case the classification task is "one out of q" and we want the model output be continuous and approximating a probability measure over $[0, 1]^q$ as close as possible (in the Kullback-Liebler sense) to the one of the training set. Hence, a proper function is the mean cross-entropy between the labels and the model outputs [12]

(3)
$$E_D(\mathbf{w}) = \frac{1}{Nq}\sum_{k=1}^{N}\sum_{l=1}^{q} o_l^{(k)} \log \frac{o_l^{(k)}}{y_l^{(k)}} + (1 - o_l^{(k)}) \log \frac{1 - o_l^{(k)}}{1 - y_l^{(k)}} ,$$

- The log priors $E_W(\mathbf{w}) = \alpha log P(\mathbf{w}_i|\mathcal{H}_i)$ on the weights. This term expresses prior knowledge about the weights, and acts as a Tichonov regularizer [62] applied to E_D. The necessity of a regularizer is due to the fact that, during training, the parameters are influenced by the noise on the data as well, thus making an optimum based on E_D alone highly sensitive to a particular training set. The net outcome would then be a network with poor classification capabilities when tested on a different (but identically distributed) test data set. With soft-threshold hidden units and output nodes, a common choice for the regularizer is a quadratic form:

(4)
$$E_W(\mathbf{w}) = \frac{1}{2}\sum w_i^2 ,$$

  which prevents the connection weights from growing too much, thus saturating the nodes.

It is then

(5)
$$M(\mathbf{w}, \alpha) = E_D(\mathbf{w}) + \alpha E_W(\mathbf{w}) ,$$

and the linear combination weight α measures the amount of regularization introduced by the prior. Its value must be chosen so that the optimum $\mathbf{w}$ is (ideally) determined only by the information present in the data plus the prior, with minimal influence of the noise. This is accomplished by optimizing, along with M, the "evidence" in the data for α, regardless of the parameters $\mathbf{w}$, i.e. the marginal

(6)
$$\int d\mathbf{w} P(\{D\}|\mathbf{w}, \mathcal{H}) P(\mathbf{w}|\mathcal{H}) ,$$

The proper optimization rule for α can then be derived analyzing the properties of the evidence at its optimum [54].

Testing

In order to efficiently *use* an already trained classifier to make predictions about which object corresponds to a particular model, we must also account for the uncertainty in the model parameters. In the case of significant model error bars $\mathbf{C}$, the classifier output to the most probable model $\mathbf{m}$ is generally biased, i.e. *not* a good representative of the class for the posterior distribution of possible models $\boldsymbol{\mu}$ given the data, $P(\boldsymbol{\mu}|\{D\})$, even for $\boldsymbol{\mu}$ in the neighborhood of $\mathbf{m}$.[6] The correct approach is to treat the models as nuisance parameters to

[6] Technically, this is a consequence of the normalization of probabilities, which causes the mapping from models to class probabilities to be necessarily nonlinear.

be "marginalized away", so to expose the data's support for each class:

$$P(o_i|\{\mathbf{x}\}) = \int d\boldsymbol{\mu}\, P(o_i|\boldsymbol{\mu})\, P(\boldsymbol{\mu}|\{\mathbf{x}\}) \quad , \tag{7}$$

where $P(o_i|\boldsymbol{\mu})$ is given by the trained classifier's output, and the integral is extended over model space. This integral can be computed using the following approximations:

- The model posterior $P(\boldsymbol{\mu}|\{\mathbf{x}\})$ is locally approximated, in the neighborhood of the optimal fit $\mathbf{m}$, with a Gaussian with mean $\mathbf{m}$ and covariance matrix $\mathbf{C}$.
- The classifier's function $\mathbf{a}(\cdot)$ is locally linearized in the neighborhood of $\mathbf{m}$, i.e. $\mathbf{a}(\boldsymbol{\mu}) \simeq \mathbf{a}(\mathbf{m}) + \mathbf{g}_\mathbf{a}(\boldsymbol{\mu} - \mathbf{m})$, where $\mathbf{g}_\mathbf{a} \triangleq (\partial\mathbf{a}/\partial\boldsymbol{\mu})_{\boldsymbol{\mu}=\mathbf{m}}$ is the sensitivity of $\mathbf{a}(\cdot)$ computed at $\mathbf{m}$.

These imply that the distribution of the $\mathbf{a}(\cdot)$ values can be locally approximated, in a the neighborhood of $\mathbf{a}(\mathbf{m})$, by a Gaussian $N(\mathbf{a}; \mathbf{a}_m, \mathbf{C}_a)$, with mean $\mathbf{a}_m \triangleq \mathbf{a}(\mathbf{m})$ and covariance matrix $\mathbf{C}_\mathbf{a} \triangleq \mathbf{g}_a^T\, \mathbf{C}\, \mathbf{g}_a$, yielding for the marginalized prediction,

$$P(o_i|\mathbf{m}, \mathbf{C}) \triangleq P(o_i|\{\mathbf{x}\}) \simeq \int d\mathbf{a}\, f_i(\mathbf{a})\, N(\mathbf{a}; \mathbf{a}_m, \mathbf{C}_a)\,, \tag{8}$$

the integral being now extended over $\mathbf{a}$ space. The symbol $P(o_i|\mathbf{m}, \mathbf{C})$ for the marginal, although an abuse of notation ($\mathbf{m}$ and $\mathbf{C}$ are not random variables), is used to remark that the approximation can be computed using model-related quantities only. This approximation is adequate as long as $\mathbf{a}(\cdot)$ does not vary too rapidly within the model's confidence interval. The integral of a logistic (or softmax) function times a Gaussian cannot be solved in closed form, though it can easily be evaluated numerically. However, a well behaved approximation has been given [76] for the logistic case (i.e. binary classification), from which useful insight can be drawn,

$$P(o_i|\mathbf{m}, \mathbf{C}) \simeq \phi(a_m, s^2) \triangleq f\left(a_m\; k(s^2)\right) \quad , \tag{9}$$

where $s^2 = \mathbf{g}_a^T\, \mathbf{C}\, \mathbf{g}_a$ is the (scalar, in this case) variance of $a(\cdot)$, and

$$k(s^2) \triangleq \left(1 + \pi s^2/8\right)^{-\frac{1}{2}} \quad . \tag{10}$$

This result is intuitively satisfactory: $k(s^2) \in [0, 1]$, hence the uncertainty in the input reduces the confidence in the prediction by "pulling" the argument of $f(\cdot)$ toward 0, and hence the estimated class probability toward $\frac{1}{2}$. This is demonstrated in Figure 4 (left).

The above approximation falls graciously for $a^2 \gg s^2$, and is formally identical to MacKay's [55]. The difference is in the uncertain variables the marginalization is made over: the classifier parameters $\mathbf{w}$ in his case, the classifier inputs $\mathbf{m}$ in ours.

3.5. **Pose and Point Estimation.** The next logical step, after modeling and recognition of object parts, is the estimation of the position and pose of the object. This can be accomplished by taking advantage of a "nominal" (often CAD/CAM) model for the recognized object, which is then positioned in space (the sensor's space) by matching *in space* the recognized parts.

As explained above, we regard the estimation of pose and points as a unique problem: the reason for determining object pose is finding where some relevant "target" object points are, and the pose itself is determined by identifying and positioning particular "landmark" object

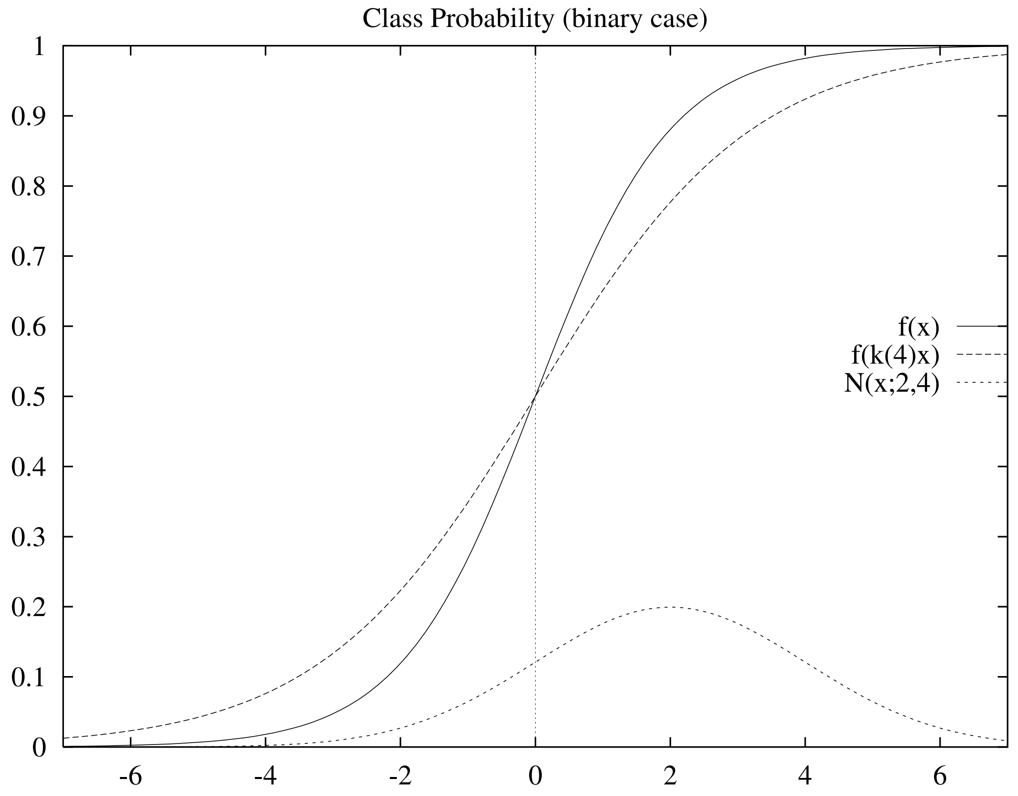


FIGURE 4. Left: examples of marginalized and non-marginalized predictions in the binary case for $s^2 = 4$.

points. The information flow and processing stages we refer to in our technique are then those depicted in Figure 5.

Thus, starting from rather general considerations of rigid body kinematics, and using suitable approximations, we derive an expression for the predictive distribution of the object points, given any three of them and their associated uncertainty. To do so we describe how such three "landmark" points can be determined by recognizing object parts from geometrical models of them, and the above uncertainties expressed. Finally, we show how the models can be built from the imaged data.

From classic kinematics of rigid bodies it is known that the knowledge of a triangle LMN rigidly attached completely specifies a rigid body in space (see Figure 6). This is because every other "target" point Q of the body is related to LMN by:

$$Q(L, M, N) = L + l\overrightarrow{LM} + m\overrightarrow{LN} + n(\overrightarrow{LM} \times \overrightarrow{LN}) \ , \tag{11}$$

where $L \triangleq \overrightarrow{OL}$, $M \triangleq \overrightarrow{OM}$ and $N \triangleq \overrightarrow{ON}$ are the locations of the triangle vertices with respect to the origin O of the reference frame (e.g. the sensor's one), and l, m, n are the projections of the $\overrightarrow{LQ}$ vector on $\overrightarrow{LM}$, $\overrightarrow{LN}$ and $(\overrightarrow{LM} \times \overrightarrow{LN})$ respectively.[7] The l, m, n components will be read on a "nominal" object model, e.g. a CAD model for manufactured parts, or a model inferred from prior accurate measurements. In either case we shall assume that the

[7]Two oriented triangle sides define an object frame, whose attitude w.r.t. the reference frame is, by definition, *the* solution of the pose estimation problem. The object frame may be orthogonalized by Gram's

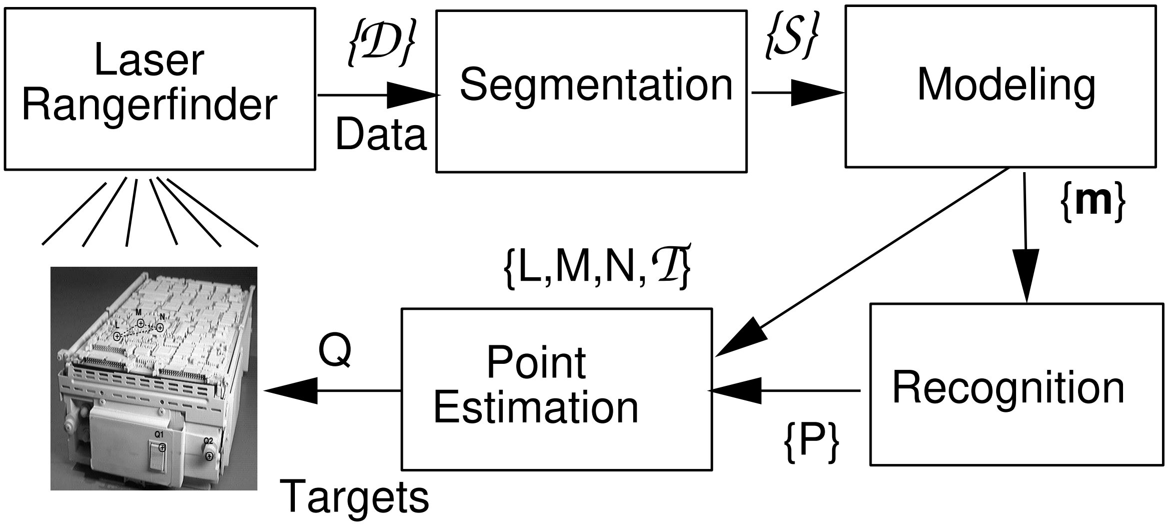


FIGURE 5. A general scheme for the proposed pose/point estimation technique. Refer to the text for the exact meaning of the various symbols

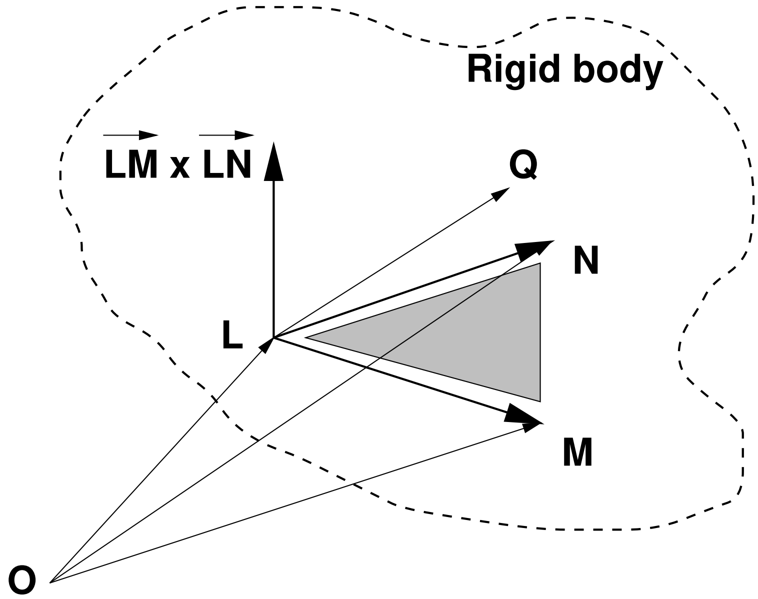


FIGURE 6. Pose specification for a rigid body, given a triangle LMN of landmark points on it.

uncertainty on l, m, n is negligible w.r.t. the one on the landmarks' position, i.e. the one involved in *using* the nominal model for making predictions about the object pose.

---

algorithm. Orthogonalization, however, would introduce further errors and unnecessary complications, without adding any insight to the treatment, hence we shall systematically refer to general (i.e. non-orthogonal frames).

Equation 11 indicates that the pose/point estimation problem for a rigid body is solved once a minimum of three "landmark" points have been identified on the body itself and positioned in space This, however, does not account for the uncertainty in the landmarks, and for its propagation upon the estimated target Q. We are thus led to regard the positions L, M and N of the landmarks as random variables to be inferred from image data $\{\mathbf{x}\}$, used to predict the position of interesting target points, and eventually marginalized away to expose the data's support for the prediction itself. The predictive target distribution

$$P(Q|\{\mathbf{x}\}) = \int d(L, M, N)\, P(Q|L, M, N)\, P(L, M, N|\{\mathbf{x}\}) \ , \tag{12}$$

where the integral is extended over the whole 9-dimensional (L,M,N) space, is then to be be regarded as *the* solution of the pose/point estimation problem. The actual computation of the distribution via the above equation is a daunting task in all but the most trivial cases: our analysis will then follow a Bayesian MAP approach, the aim being the determination, through suitable approximations, of the distribution maximum and of the predictive errorbars around it.

3.5.1. *Point estimation.* A closed-form evaluation of the conditional $P(Q|L, M, N)$ via Equation 11 is not easily obtained, since that expression is nonlinear in the landmark coordinates. For the purposes of the present work, we shall then use the following approximations: (a) A *local* linear approximation of Equation 11 around the MAP locations $(\hat{L}, \hat{M}, \hat{N})$ of the triangle vertices. (b) A *local* Gaussian approximation for the landmark posterior $P(L, M, N|\{\mathbf{x}\})$, with mean at $(\hat{L}, \hat{M}, \hat{N})$ and covariance matrix C_{LMN} (9×9 dimensional) to be determined.[8]

These assumptions together imply a local Gaussian approximation of $P(Q|\{\mathbf{x}\})$, whose mean is given by Equation 11, computed at $(\hat{L}, \hat{M}, \hat{N})$, and whose (3×3) covariance matrix is given by

$$\mathbf{C}_Q = \left(\frac{\partial Q(L, M, N)}{\partial (L, M, N)} \right)' \mathbf{C}_{LMN} \frac{\partial Q(L, M, N)}{\partial (L, M, N)} \ , \tag{13}$$

where $(\partial Q(L, M, N)/\partial(L, M, N))$ is the (9×3) Jacobian matrix of the 3-D coordinates of $Q(L, M, N)$ w.r.t. the landmarks, computed from Equation 11 at $(\hat{L}, \hat{M}, \hat{N})$.

3.5.2. *Joint landmark estimation.* We now face the problem of determining the position of the fiducial points L, M, N, along with their own posterior distribution, at least in the above Gaussian approximation. This is obviously a joint recognition and estimation problem: the landmarks are positioned by identifying them among all sensed data. Or rather, since the "true" landmarks will not generally be among the data (if nothing else, because of measurement noise), their locations must be inferred after identifying in the data other suitable (large, reliable) features segmented within the data. This is the same as saying that each of the landmarks, e.g. L, should really be fully qualified as a 3-tuple $(L, \mathcal{S}_L, \mathcal{P}_L)$, consisting of a position L, of a segment of data $\mathcal{S}_L$ within which L is likely to be located, and

[8] We stress the *locality* of these assumptions. While most of the models used in our exposition are in fact nonlinear, a local linear approximation of them around relevant points may prove practically adequate. Likewise, since the models are obtained by MAP estimation on dense data sets, a local Gaussian approximation of their posterior distribution will often prove adequate, on account of the Central Limit Theorem and of the quadratic behavior of nondegenerate cost functions in the neighborhood of their optima. The experiments decide of the validity of both assumptions.

of a segment label $\mathcal{P}_L$ identifying $\mathcal{S}_L$ as that particular segment (the "part") containing the landmark L. However, not to clutter the following formulae, we shall still use the previous expression for the landmark posterior as an abuse of notation: it will be evident from context whether we refer to the fully qualified landmarks rather than to just their locations.

We then proceed by considering separately, within the landmark recognition problem,

(a) The mutual relationship of the landmarks, i.e. that they belong to a triangle $\mathcal{T}$ whose side lengths and orientation[9] are supposedly known from a nominal object model: only the landmark locations matter for this relationship;
(b) The location of the individual landmarks: the full qualification of the landmarks is obviously relevant here.

In other words, we propose to select the most likely triplet of landmarks, among all possible candidates, by enforcing the mutual constraint $\mathcal{T}$. This will be done by maximizing

$$P(L, M, N, \mathcal{T}|\{\mathbf{x}\}) = P(\mathcal{T}|L, M, N)\, P(L, M, N|\{\mathbf{x}\}) \ , \tag{14}$$

over the set of all possible triplets of candidate landmarks.[10] The term $P(\mathcal{T}|L, M, N)$ is the likelihood of landmark frame, and we model it in a form that considers separately the contributions due to errors in estimating the triangle side lengths ($E_{\mathcal{L}}$) and orientation ($E_{\mathcal{O}}$):

$$P(\mathcal{T}|L,M, N) \propto \mathrm{e}^{-\beta_{\mathcal{L}} E_{\mathcal{L}}(L,M,N)} \times (1 - E_{\mathcal{O}}(L, M, N)) \ . \tag{15}$$

For the first error term we use a standard least squares formulation:

$$E_{\mathcal{L}} = (\overline{LM} - \widehat{LM})^2 + (\overline{LN} - \widehat{LN})^2 + (\overline{MN} - \widehat{MN})^2 \tag{16}$$

where $\overline{LM}$ and $\widehat{LM}$ are respectively the nominal and estimated lengths of LM, and similarly for the other triangle sides. For the second error term we can use a stronger constraint, since we know on which side of the object (i.e. of the LMN plane) the sensor is supposed to be. We then make the triangle likelihood vanish if the estimated and nominal orientations disagree:

$$E_{\mathcal{O}} = \begin{cases} 1 & \text{if } (LM \times LN) \cdot (\hat{L}\hat{M} \times \hat{L}\hat{N}) < 0, \\ 0 & \text{otherwise} \ . \end{cases} \tag{17}$$

The term $P(L, M, N|\{\mathbf{x}\})$ in Equation 14 refers to the inference of the landmarks *per se*, and we make for it an independence assumption

$$P(L, M, N|\{\mathbf{x}\}) = P(L|\{\mathbf{x}\})\, P(M|\{\mathbf{x}\})\, P(N|\{\mathbf{x}\}) \ . \tag{18}$$

As we'll see shortly, this assumption is motivated by the fact the the landmarks are inferred from separate models for the data segments.

---

[9]The orientation of a triangle is, by definition, the parity of the frame defined by its oriented sides in a given order.

[10]In a rigorous Bayesian approach we should *marginalize* w.r.t. $\mathcal{T}$. The location of the maximum yields a good approximation of the marginal whenever the distribution is "tight" enough around it. Note that if the segmentation procedure used yields $n_{\mathcal{S}}$ segments, the search for the maximum is done over a set of $\binom{n_{\mathcal{S}}}{3} \sim O(n_{\mathcal{S}}^3)$ triplets, assuming one landmark per segment. A robust segmentation procedure [48] can keep this size quite manageable in the cases encountered so far.

3.5.3. *Individual landmark estimation.* The individual landmark points are located using geometric models of the data segments they belong to, and identified via model-based part recognition. The use of shape models for the parts is an asset, since it reduces the error in each estimated landmark's position through regression over a *dense* set of points: the points sensed on the segment the landmark belongs to. Moreover, well-selected models can adequately capture in their parameters shape features useful for identifying the parts and, *a fortiori*, the landmarks on them.[11]

The procedure can be summarized as follows. Data segments $\{\mathcal{S}\}$ are "carved" in the sensed data $\{\mathbf{x}\}$ by a segmentation procedure, and their shape is modelled by parametric regression on given 3-D shape primitives $\mathcal{M}(x, y, z; \mathbf{m})$ (e.g. blocks, cylinders, superellipsoids, etc.). The so recovered model parameters $\mathbf{m}$ act as view-independent shape descriptors, which index trough a classifier a database of labelled part models $\{\mathcal{P}\}$. Once a segment-part match is found, a corresponding landmark position (L, M or N) is determined on the matched model. Again, the associated distributions must be accounted for: the shape parameters are affected by a posterior uncertainty $P(\mathbf{m}|\mathcal{S}, \{\mathbf{x}\})$, which in turn induces a quantifiable ambiguity $P(\mathcal{P}|\mathbf{m}, \mathcal{S}, \{\mathbf{x}\})$ in the indexing, and both sources of error are to be carried through the analysis.

From a Bayesian point of view, the models are auxiliary (nuisance) variables, which must be marginalized away in order to expose the data's support for each landmark estimation. So, for the location of landmark L we have:

$$P(L|\{\mathbf{x}\}) \triangleq P(L, \mathcal{S}_L, \mathcal{P}_L|\{\mathbf{x}\}) = \int d\mathbf{m}\, P(L, \mathcal{P}_L, \mathcal{S}_L, \mathbf{m}|\{\mathbf{x}\}) \ , \tag{19}$$

and similarly for the other landmarks, where $(L, \mathcal{S}_L, \mathcal{P}_L)$ indicates the event in which the landmark is correctly identified and located, and the integral is extended over all model parameter space.

A particular inference procedure is just a way to compute the above marginal. In our case, we infer landmarks from models, and models and parts from segments, hence we write:

$$\begin{aligned} P(L, \mathbf{m}, &\mathcal{P}_L, \mathcal{S}_L|\{\mathbf{x}\}) = \\ &P(L|\mathbf{m}, \mathcal{P}_L, \mathcal{S}_L, \{\mathbf{x}\})\, P(\mathbf{m}|\mathcal{P}_L, \mathcal{S}_L, \{\mathbf{x}\}) \\ &\times P(\mathcal{P}_L|\mathcal{S}_L, \{\mathbf{x}\})\, P(\mathcal{S}_L|\{\mathbf{x}\}) \ . \end{aligned} \tag{20}$$

But landmarks are computed on models, hence they do not depend on segments, parts or data once the model is given. Similarly, models are fitted to segments, and parts are recognized from models fitted to segments, so the above simplifies to

$$\begin{aligned} P(L, \mathcal{S}_L, \mathcal{P}_L, \mathbf{m}|\{\mathbf{x}\}) = P(L|\mathbf{m})\, P(\mathbf{m}|\mathcal{S}_L)\, P(\mathcal{S}_L|\{\mathbf{x}\}) \\ \times \int d\mathbf{m}\, P(\mathcal{P}_L|\mathbf{m})\, P(\mathbf{m}|\mathcal{S}_L) \ , \end{aligned} \tag{21}$$

where the last integral represents $P(\mathcal{P}_L|\mathcal{S}_l, \{\mathbf{x}\}) = P(\mathcal{P}_L|\mathcal{S}_l)$, as determined through modeling ($P(\mathbf{m}|\mathcal{S})$) and model-based recognition ($P(\mathcal{P}_L|\mathbf{m})$). Substituting this into Equation 19 we

[11] Of course, implicit in all this is the assumption that the objects at hand are amenable to segmentation according to a chosen solid shape theory, that a chosen class of models is competent to represent the segments themselves, and that the model parameter distributions for different segments are separated enough for the purposes of recognizing them as parts. See [13, 14, 48, 86] for discussions of these issues.

get the final expression for each landmark posterior prediction:

$$
\begin{aligned}
P(L|\{\mathbf{x}\}) = \Big[\, P(\mathcal{S}_L|\{\mathbf{x}\}) \int & d\mathbf{m}\, P(L|\mathbf{m})\, P(\mathbf{m}|\mathcal{S}_L) \\
& \times \int d\mathbf{m}\, P(\mathcal{P}_L|\mathbf{m})\, P(\mathbf{m}|\mathcal{S}_L) \Big] \,,
\end{aligned} \tag{22}
$$

and here the first integral is simply the landmark conditional given the segment, $P(L|\mathcal{S}_L)$, as determined through modeling.

3.5.4. *Practical implementation.* In the implementation demonstrated in this paper, the terms in Equation 22 are computed as follows:

1. A range image of the object is segmented by non-linear diffusion of the negative curvature minima [48]. Segments with less data points than a pre-determined threshold are rejected (since we look for large, reliable parts). For the moment, the remaining $n_{\mathcal{S}}$ segments are assigned a uniform probability distribution $P(\mathcal{S}|\{\mathbf{x}\}) \triangleq 1/n_{\mathcal{S}}$.
2. A MAP estimate $\widehat{\mathbf{m}}$ of the parameters of the model fitting each segment is determined by optimizing a suitable cost function depending on $\mathbf{m}$ and the data. In our case, the models $\mathcal{M}$ are superellipsoids fitted to the segmented data [86] by minimizing the sum of the squared radial distances of the data from the model, over a noninformative prior for the model parameters. As in standard least squares regression, a local Gaussian approximation for the model posterior $P(\mathbf{m}|\mathcal{S})$ is made in the neighborhood of the optimum, with mean equal to $\widehat{\mathbf{m}}$, and covariance matrix $\mathbf{C}_{\mathbf{m}}$ equal to the inverse of the Hessian matrix of the cost function, computed at the optimum.
3. A trained discriminative classifier then produces *online* an estimate of $P(\mathcal{P}_L|\mathbf{m})$, $P(\mathcal{P}_M|\mathbf{m})$ and $P(\mathcal{P}_N|\mathbf{m})$. The estimate is marginalized with respect to the model uncertainty $P(\mathbf{m}|\mathcal{S})$ as described in [13, 14].
4. The model is also used to infer the MAP location of the landmark given the segment: for this purpose we simply use one of the superellipsoid vertices. The conditional landmark distribution given the segment is again obtained from a local linear approximation of the model in the neighborhood of the optimum, yielding a local Gaussian approximation for $P(L|\mathcal{S})$ with mean $\mathcal{M}(L;\widehat{\mathbf{m}}) \triangleq \mathcal{M}(x_L, y_L, z_L;\widehat{\mathbf{m}})$ and covariance matrix:
$$
\mathbf{C}_L = \left(\frac{\partial \mathcal{M}(L;\widehat{\mathbf{m}})}{\partial \mathbf{m}}\right)' \mathbf{C}_{\mathbf{m}}\, \frac{\partial \mathcal{M}(L;\widehat{\mathbf{m}})}{\partial \mathbf{m}} \tag{23}
$$
and similarly for M and N.
5. The above steps are repeated for all segments, thus determining all candidate landmark locations and identifications, and consequently all candidate triangles. Finally, the triangle with the highest likelihood is found by maximizing Equation 15, yielding a MAP landmark estimate and its predictive errorbars [15].

The classifier used in step 3 is trained *offline* (i.e. once and for all, for a given set of objects), by iterating steps 1 and 2 for several views of each segment. The recovered models are manually labelled with the corresponding parts, so to build a model-part pair database (training set). In our case, as in [13, 14], the discriminative classifier is implemented in the form of a committee of feedforward connectionist architectures, trained and selected using MacKay's "Bayesian evidence" framework [52].

Note that, within the above approximations, the final joint landmark posterior (equal to the product of the individual landmark posteriors maximizing Equation 15) is a Gaussian with respect to spatial coordinates, as required by Equation 13. The joint landmark covariance C_{LMN} of that equation is simply the block-diagonal 9×9 matrix defined by the individual landmark posterior covariances.

Equations 11 and 13 eventually yield the MAP target prediction and its predictive errorbars.

We now have all the "tools" in their places: we have described a Bayesian framework for solving the pose/point estimation problem via model-based object recognition, taking consistently into account various sources of uncertain information.

We have also outlined a practical implementation of this technique, which depends on a local, Gaussian approximation for the part model posterior distribution. This automatically implies a local Gaussian approximation for the landmark posterior and, when joined with the quadratic form for the landmark triangle likelihood, agrees with (a) the one postulated in Equation 13 for the triangle posterior and (b) with the assumption of independence of the three landmark posterior distribution.

Our approximated implementation of the above scheme is then, at least, internally consistent. Of course, the *validity* of the approximations and assumptions in the face of real 3-D data has yet to be proved. This is the subject of the following section.

4. Experimental Demonstrations

In this Section, we present the results obtained from the implementation of our recognition and pose estimation methodology. In the first part, we will present the experimental set-up and the testing conditions. A test example will then be demonstrated in detail.

4.1. **Experimental Setup.**

4.1.1. *Range Finder.* The laser range finder used to collect 3-D measurements for these preliminary experiments has been developed in the APL (Autonomous Perception Laboratory) in McGill University. As shown in figure 7, it is composed of a video camera, a filter and a source of laser producing a plane of light. The range is measured by triangulation principle. The range operation is from 20 cm to 70 cm with a precision of about 0.3 mm at its best. A high density range image (200x200 for instance) can be acquired in about 15 sec.

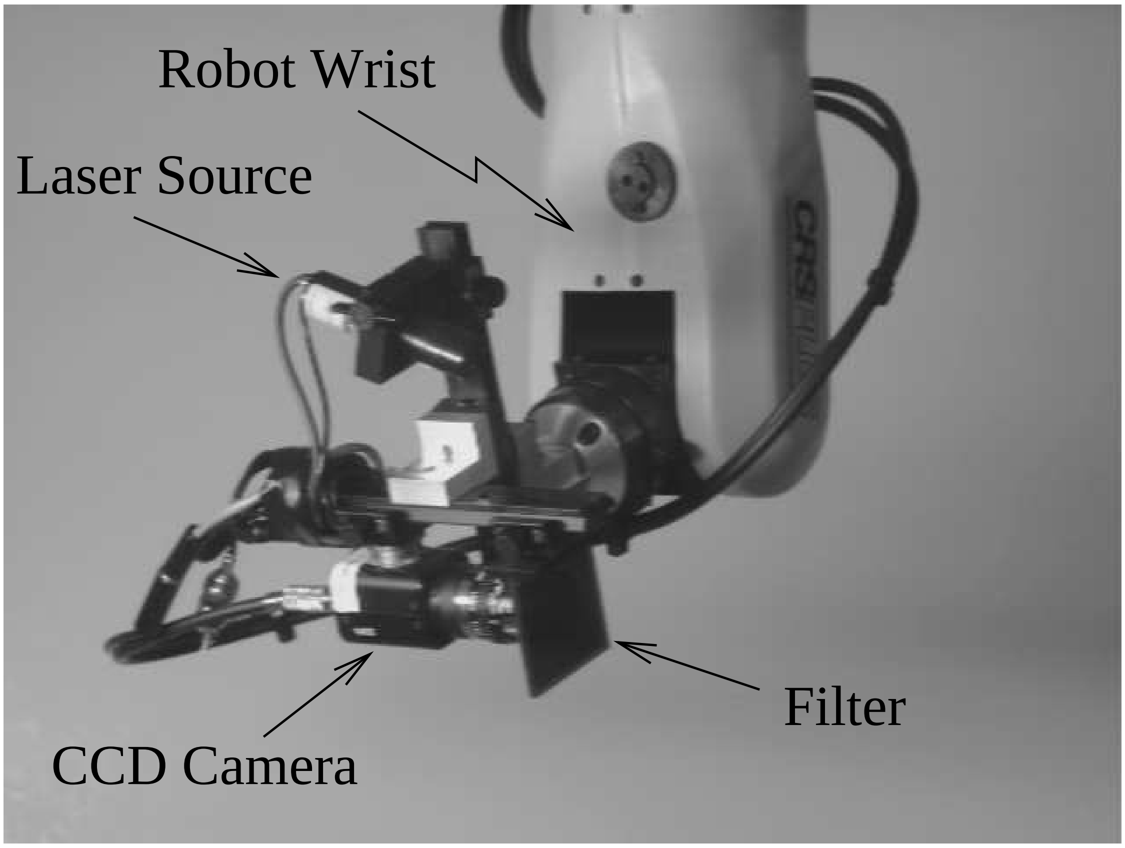


Figure 7. Laser range finder used for the experiments. A laser source projects a line of light permitting surface measurements by triangulation principle.

4.1.2. *Manipulator.* The scanner is mounted on the end-effector of a CRS Plus G3000 gantry robot (see Figure 8). This 6 DOF robot has a workspace of 1 cubic meter and a repeatability precision of 0.3 mm. Global absolute precision is variable and depends on a particular physical installation. The last joint of the robot is used to perform the scanning of objects which eliminates the need for an additional motor and reduces the payload.

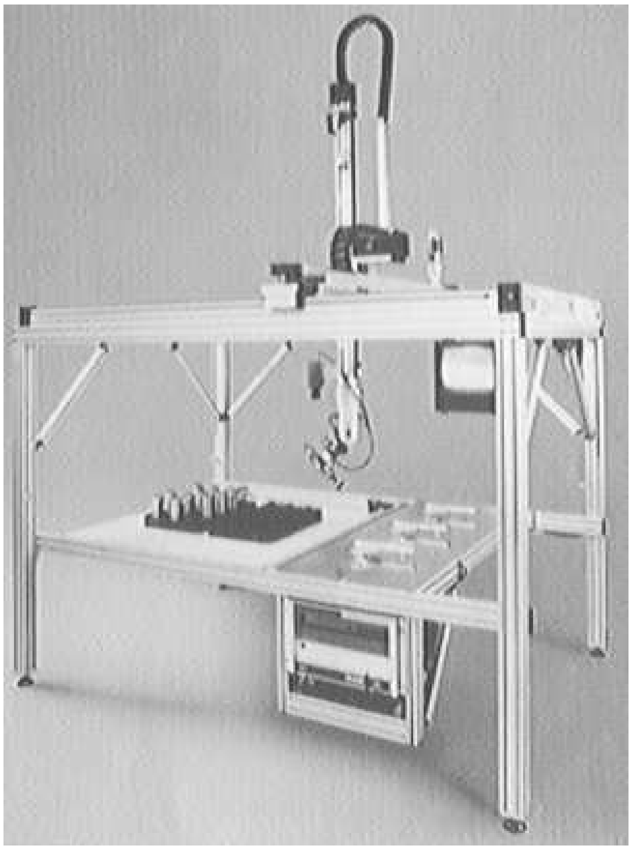


FIGURE 8. 6 DOF Gantry robot used to move the scanner around the object. Reproduced from a *CRS Plus* document.

4.1.3. *Test Object.* The object chosen for our experiments is an Apollo Hard Disk whose top cover has been removed to expose the internal electronic components. Its structural complexity is similar (up to a scale factor) to what can found in industrial environments or in space applications. It includes a large number of small components and a few large ones that can be used as landmarks to unambiguously determine the orientation and position of the object in space.

4.1.4. *Landmarks and Targets.* Figure 9 shows a close-up view of the test object. The arrows indicate the objects (landmarks) that were manually selected as landmarks. They are spatially arranged in triangle allowing the determination of a frame of reference.

Also shown are two target points whose positions must be extrapolated from the landmarks are also shown (right screw and upper right corner of the switch). They are relatively far from the landmarks and, depending on the sensor position and attitude, cannot at times be seen with them (thus the need for extrapolation).

4.1.5. *"CAD Model".* Since no CAD model of the test object was available, it was necessary to build one manually to determine the relative position between targets and landmarks. For our purpose, a very limited model is sufficient since we are only interested in the position of the various elements and not in their exact shape. Manual probing of one range image where targets and landmarks were jointly visible allowed to produce all necessary information.

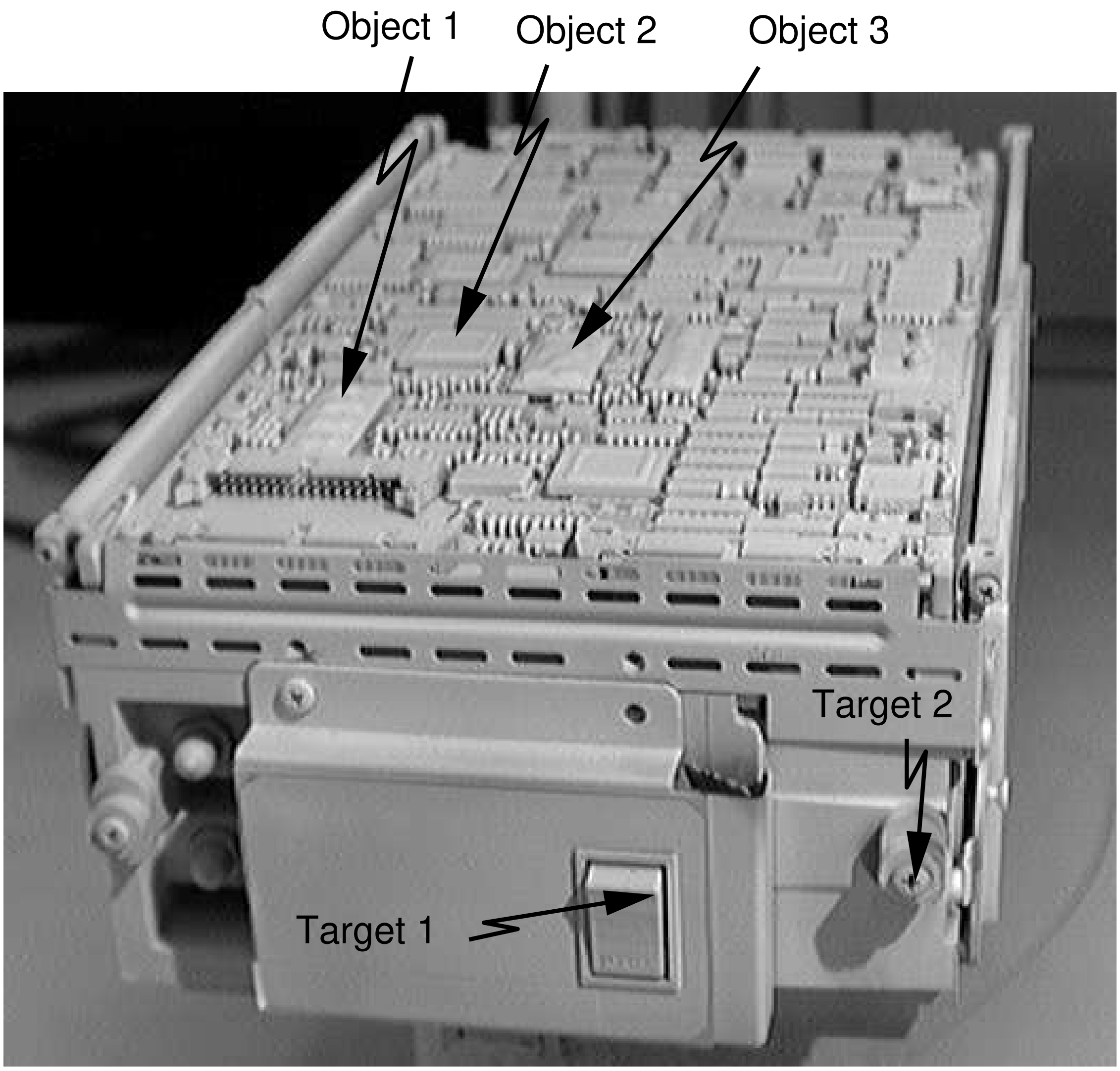


FIGURE 9. Object used to perform the proof of concept of the algorithm. The arrows indicate the selected objects (or landmarks) and the targets to be located.

4.1.6. *Base Vectors.* In our database, the position of the targets is expressed in terms of the three base vectors defined by the triangle of the landmark points. The landmarks themselves are defined as the centres of the top surfaces of the objects (integrated circuits) they belong to. The vectors are then

- v_1 : the unit length vector with same orientation as the vector going from the landmark point on Object 1 to the landmark point on Object 2.
- v_2 : the unit length vector with same orientation as the vector going from the landmark point on Object 1 to the landmark point on Object 3.
- v_3 : the unit length vector with same orientation as the cross-product of v_1 times v_2.

Using this basis, the position of the targets (T_1 and T_2) can be expressed, in object-centered coordinate frame, as:

$$T_1 = C_{11}v_1 + C_{12}v_2 + C_{13}v_3 \tag{24}$$
$$T_2 = C_{21}v_1 + C_{22}v_2 + C_{23}v_3 \quad , \tag{25}$$

the C_{ij} being the components of the target vectors in the above frame.

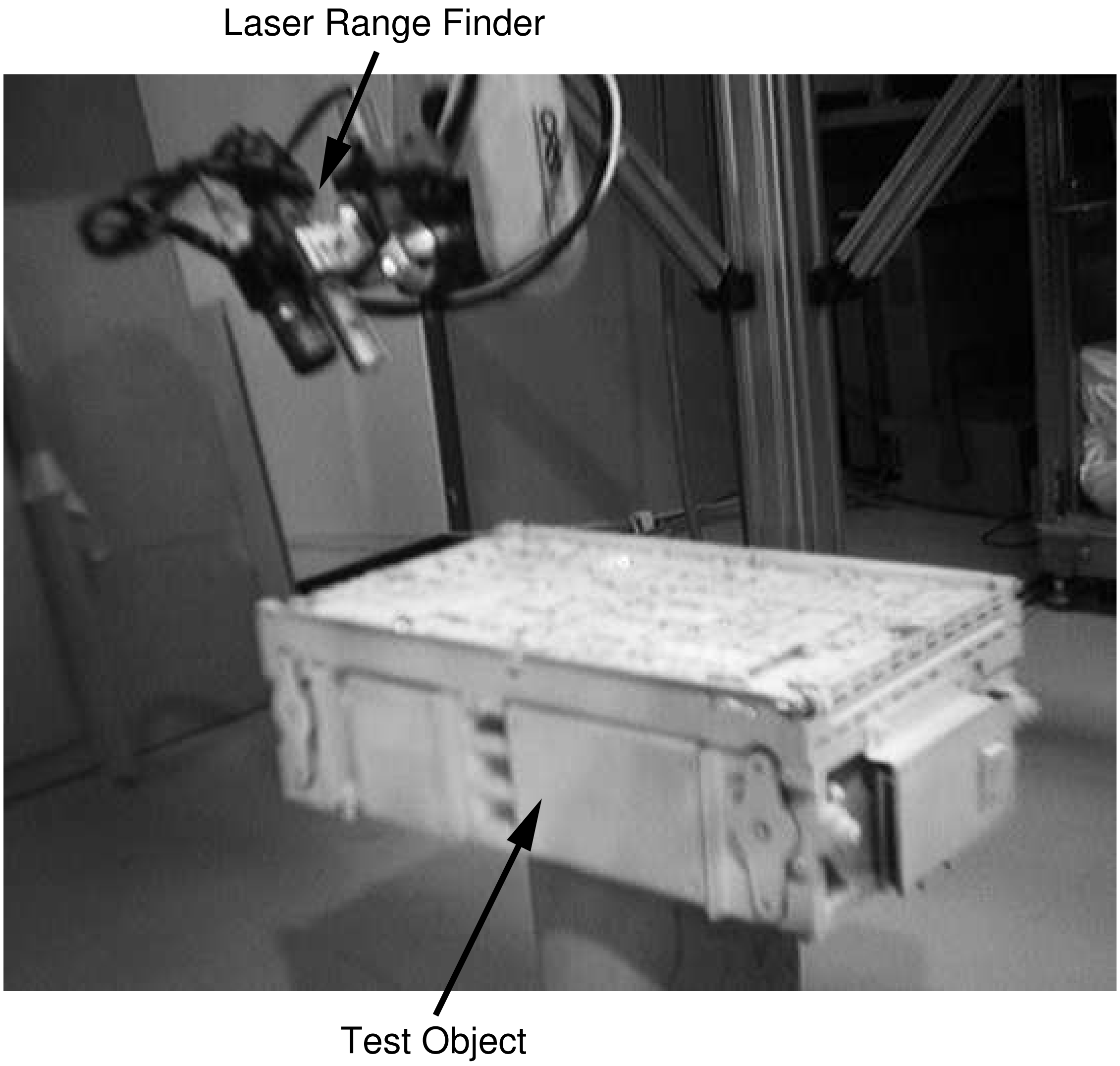


FIGURE 10. Close-up view of the set-up showing the laser range finder and the test object.

4.1.7. *Training.* Figure 10 shows a close-up of the set-up when performing a measurement. From similar configurations, twelve range images of the landmarks were acquired, segmented and fitted. The model parameters obtained for each of the landmarks were manually identified and labelled and then fed directly to the training engine. Figure 11 shows the *learning curve* of the system and its rapid convergence. Training is done off line only once and permits fast object identification involving only a few mathematical operations.

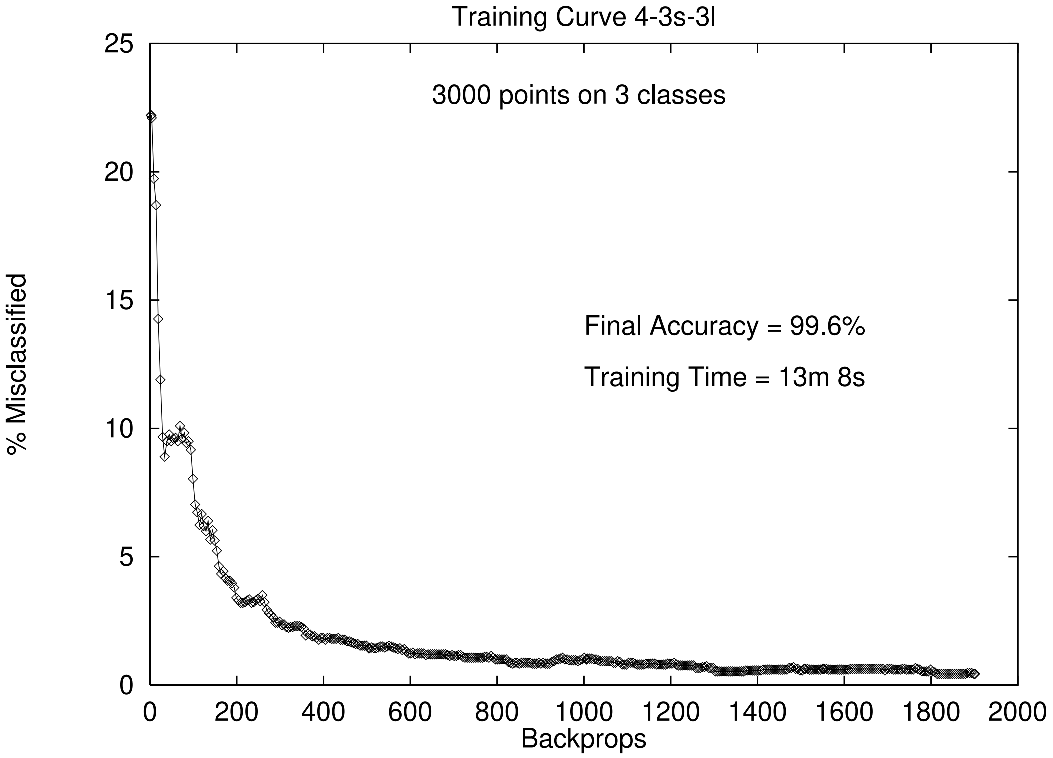


FIGURE 11. Results of the training procedure expressed as the percentage of misidentification as a function of number of back propagations.

4.2. **Results and Discussion.** Given a trained recognition system, a test range image (200 X 200) was acquired and is shown in Figure 12. The three objects (landmarks) on which the system was trained are shown by the arrows. The two targets to be extrapolated are also shown at the bottom of the image.

The result of range image segmentation is shown in Figure 13. Different colors indicate different part labels. Objects 1, 2 and 3 have been correctly segmented together with a number of other components. Because of their similar shape, the other components also found in this image are likely to be also identified as valid objects, though with possibly lower probability $P(o_L|\mathbf{m}, \{\mathbf{x}\})$. It is now evident the importance of considering their spatial arrangement as well: the triangle likelihood measure $P(\mathcal{T}|L, M, M)$ provides a further consistency check that allows to practically eliminate the "bad" candidate parts in the target prediction.

The parts found by the segmentation stage have been fitted with superellipsoids and the results are shown in Figure 14. The models appear to be rather tall because no constraint restrict their growth vertically. Since our goal is to locate the top center of the landmarks, this large vertical size has virtually no effect on the final result: the object tops are well constrained by the data, which yields very small predictive model errorbars on them.

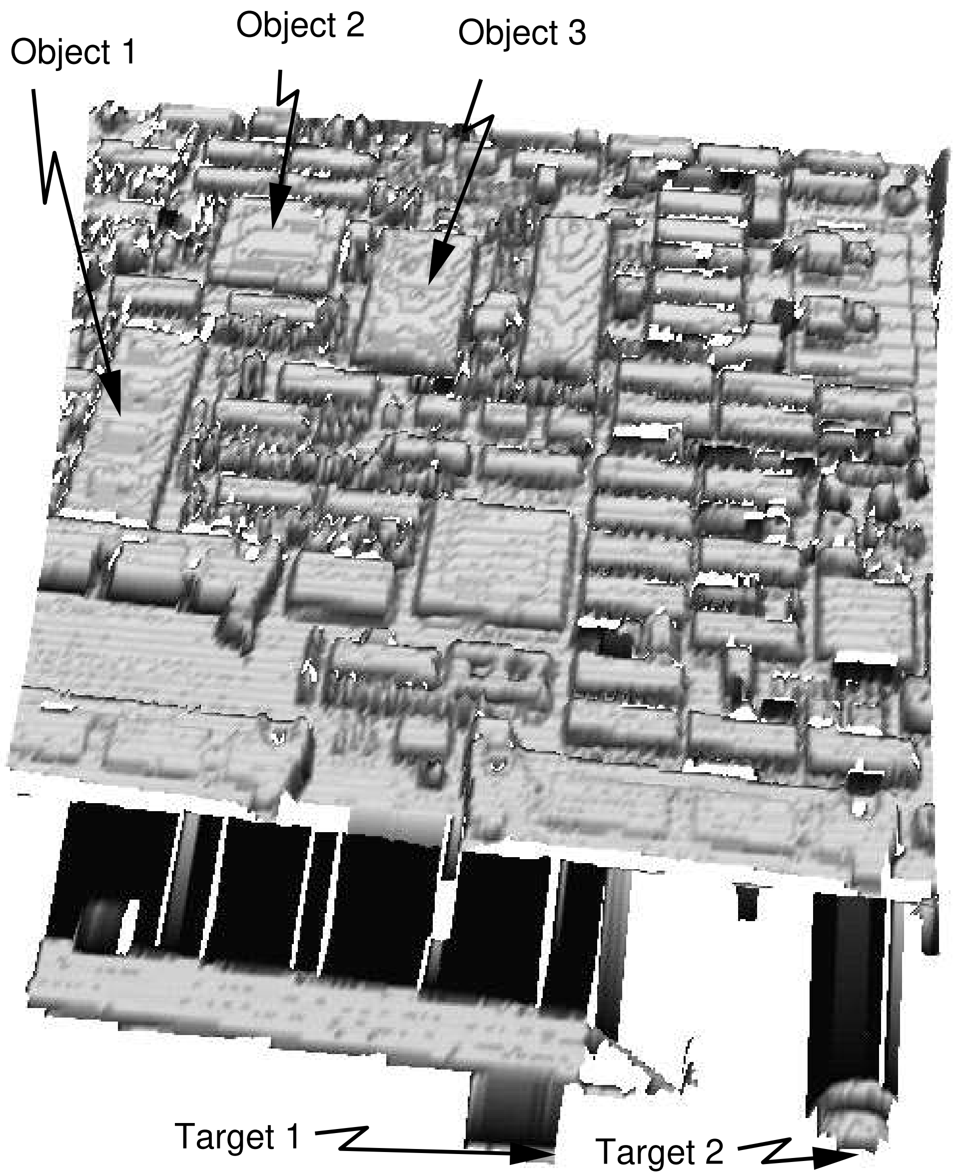


FIGURE 12. Artificial shading of the first test range image. The three objects (landmarks) and visible targets are clearly labelled.

In this case, the classifier cannot produce an unambiguous identification for object 1 (because of the similar other object immediately to the right of object 3). The similarity in the shape (and consequently in the model parameters) of these two objects explain this ambiguity. The system has also found a few twins to object 2 (e.g. the model at the extreme right in Figure 14). The combination of all these objects found produce a total of 18 possible triangles. However, the further selection criteria induced by the triangle likelihood has allowed to unambiguously select the right triplet of objects, as it can be seen in Figure 15.

Given this unique triplet of objects, an object-centered coordinate frame was determined and the target locations were successfully extrapolated. Table 1 shows a comparison between

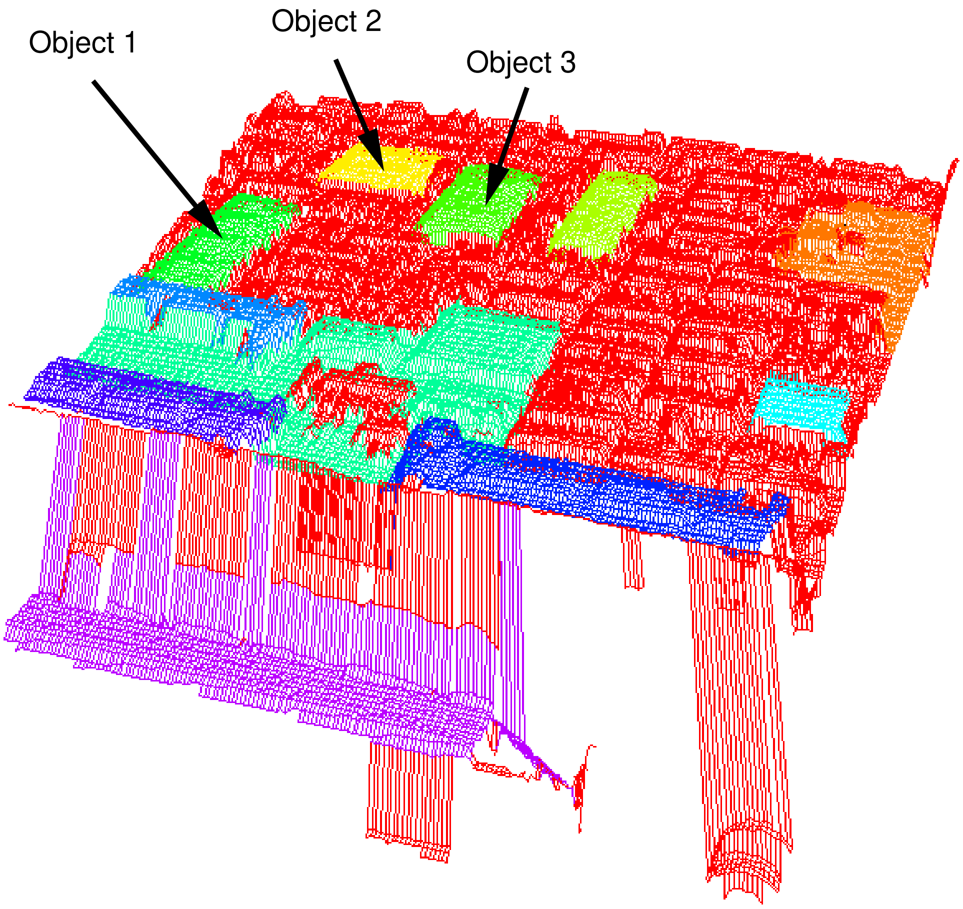


FIGURE 13. Result of the range image segmentation. Different colors indicate different part labels. Objects 1, 2 and 3 have been successfully segmented with a number of other components.

the prediction of the system and a manual probing of the targets (ground-truth). The error between the two measurements are within the two-σ interval (96% confidence level) predicted by the system.

The same results are shown graphically in Figure 16. The winner triangle have been used to extrapolate the location of the targets (blue arrows). The blue axes represent the locations found by manual probing and the gray ellipsoidal shapes represent the position and the 2-σ uncertainty found by the system. The prediction falls as expected within the tolerance interval.

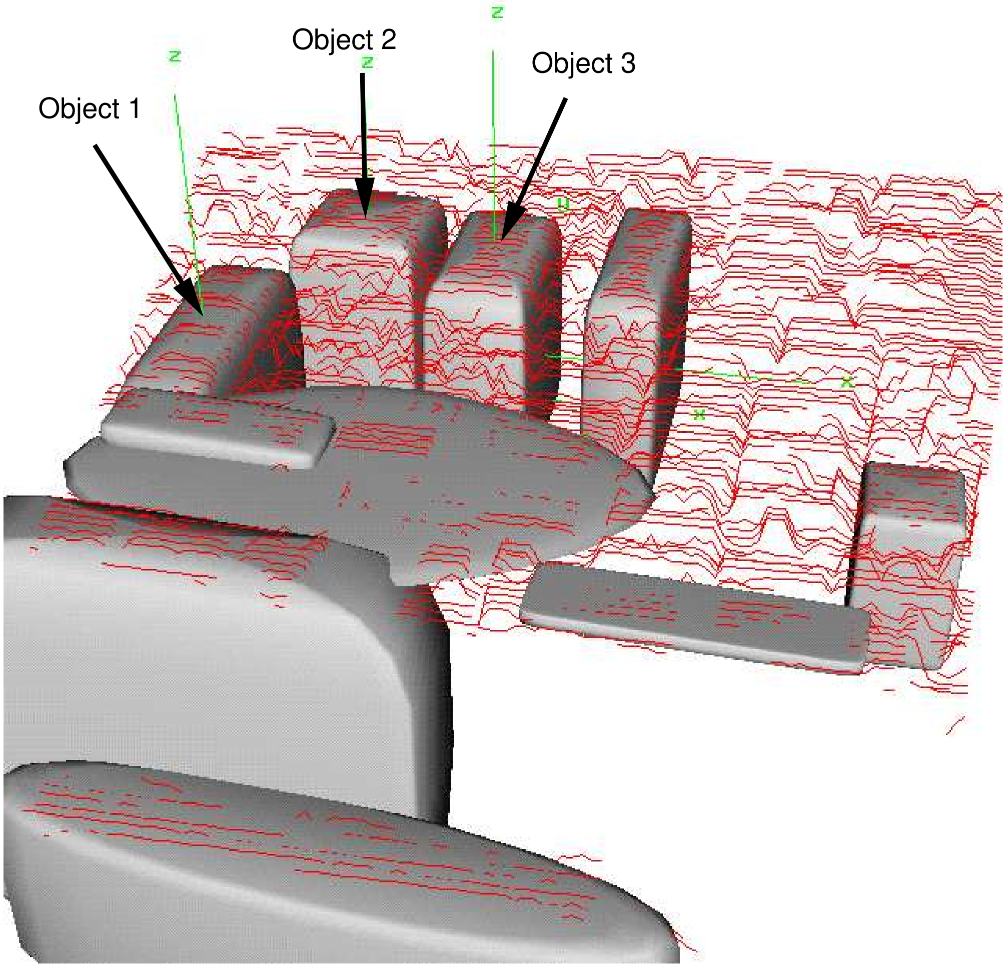


FIGURE 14. Result of the model fitting to the parts found in the segmentation stage.

	Target 1			Target 2		
Measured	408.93	394.43	28.88	462.92	407.14	25.23
Predicted	401.96	397.96	25.97	454.80	412.66	23.14
Error	6.97	-3.52	2.91	8.13	-5.51	2.09
Max. σ	3.88			4.56		

The *measured* line shows numbers obtained by manual probing of the surface. The *predicted* line shows the output of our algorithm with corresponding absolute error on the next line. *Max.* σ indicates the uncertainty determined by the method and includes raw data noise, model fit error and prediction error. Assuming Gaussian distribution, the points have 68% probability to be within one σ from the measured value and 96% probability to be within 2-σ.

TABLE 1. Results for the two targets visible in Figure 12.

5. CONCLUSION

In this report, we have introduced a novel object recognition system that is robust with respect to imaging conditions and that does not rely on any special-purpose (cooperative) landmarks. The natural parts of the object are used as landmarks that are stable with respect

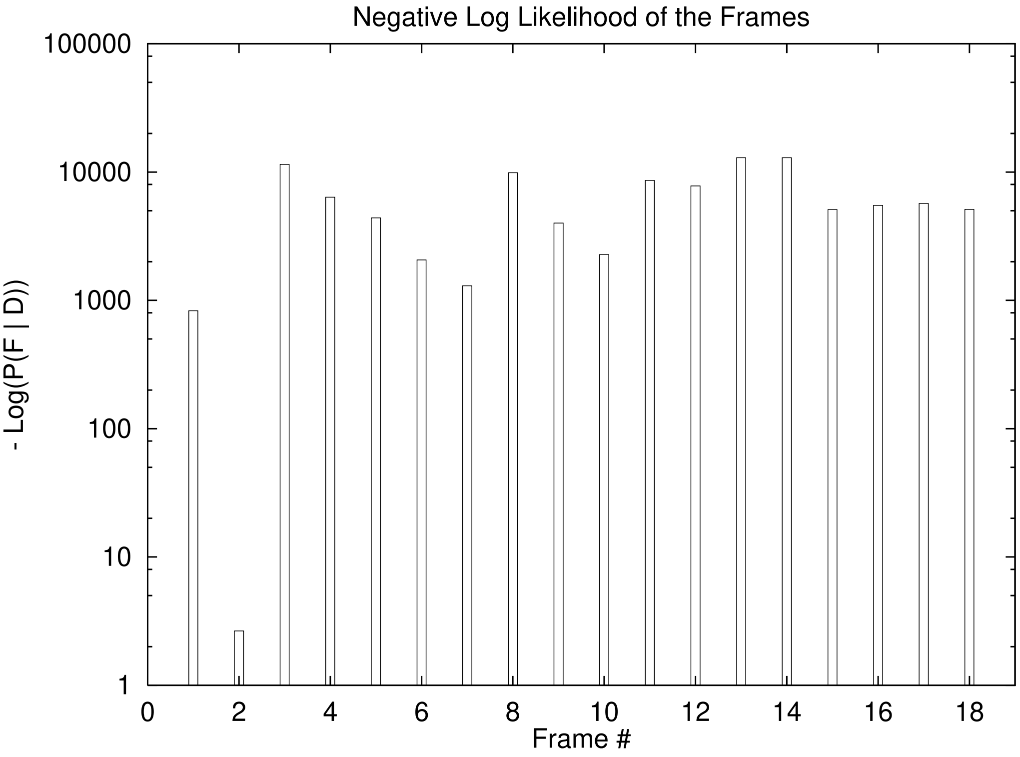
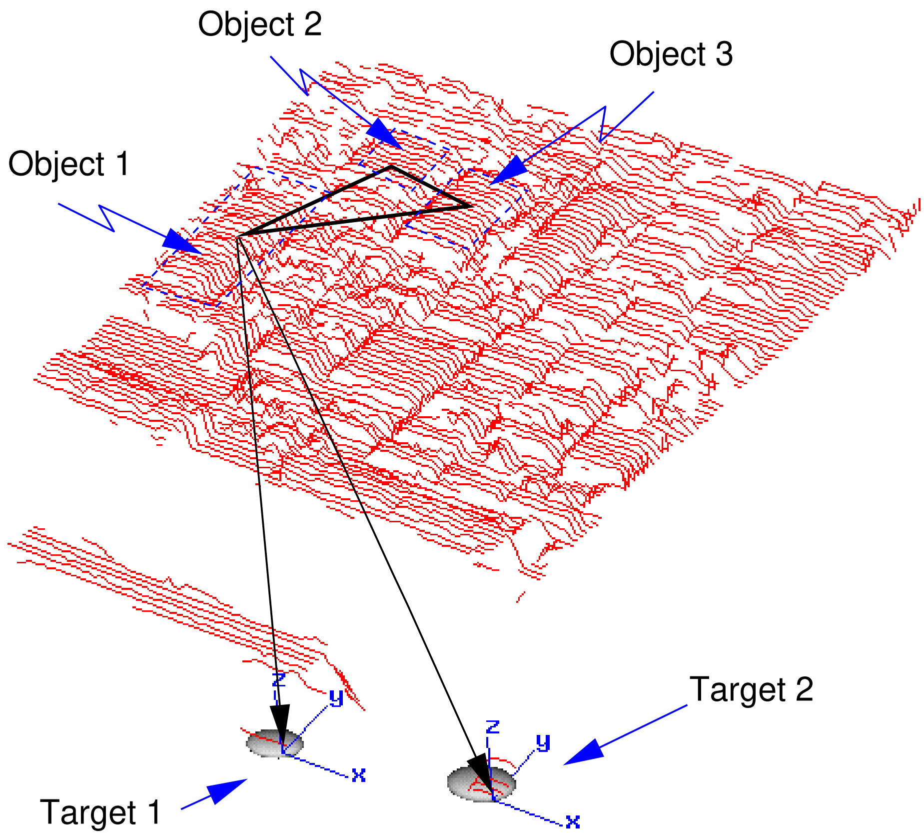


FIGURE 15. Unambiguous selection, based on the triangle likelihood, of the correct triplet of objects forming an object-centered coordinate frame. Since this is a plot of the negative log likelihood, small numbers mean high probability and *Frame 2* is a clear winner by two orders of magnitude at least. The marginalization integral is then reduced to a multiplication around the likelihood peak.

to viewpoint. A robust range image segmentation algorithm and a precise recognition system are combined to identify and locate the landmarks from which the object pose and identity can be determined.

We have shown how the automatic identification of these landmarks can be applied to the problem of object pose determination in order to build an unambiguous object-centered coordinate frame. Pose determination has been reduced to the identification of at least three landmarks. Multiple identifications are handled robustly by looking at the spatial relationship of the various parts found. A unique reference frame can be determined with a high level of confidence.

Given a model, obtained either through CAD of built interactively using the system itself, and an estimate of the pose of the object, we also proposed a deterministic framework to predict the location of possibly occluded targets on this object. The prediction was shown to be within the level of tolerance given by the system. The uncertainty associated with all



Target location as predicted by the system. The overlaid triangle represent the object-centered coordinate frame automatically found by correctly identifying the three objects. The black arrows points to the targets location in this frame as determined by our CAD model. The spatial uncertainty associated with these predictions is represented by two ellipsoids of confidence, of dimension 2-σ (96% confidence level).

FIGURE 16. Graphical interpretation of the results shown in Table 1.

the predictions computed by the system can be used to validate this information and decide whether additional data collection is warranted to further refine parameter estimates.

As a proof of concept of this approach, the work done here has raised a number of points to be examined in more detail in Phase II of this Stear project. First, a formal analysis of the absolute precision to be expected from such a system and the identification of the main factors influencing it has to be performed (tolerable sensor noise level, scalability of the approach for large structures,...). Also, it might be worthwhile to consider other types of parametric models that could be more appropriate in task-specific domains.

The Offline part of the system, namely the training, would have to be studied from a practical point of view to determine what type of parts on an object are good candidates as reliable and stable landmarks. This information would be used to design a procedure for extracting this information with minimal operator supervision, leading eventually to a fully-automated system.

When applied in very constrained environment such as a space applications, it will also be necessary to automatically choose the optimal viewpoint from a limited number of possibilities available. Such a feature could lead to significant benefits in time-critical applications.

Finally, a natural extension to this work is the addition of its dynamic counter-part: real-time object tracking. We believe that the static pose determination solution presented here can be used to initialize a tracking process that could take advantage of the all the information found in the static phase. A priori information from the precise static pose determination would be used to feed-forward the tracking process using in an efficient manner everything that is known about the object.

References

[1] T. Arbel. Recognizing volumetric objects in the presence of uncertainty. Master's thesis, McGill University, Montréal, Québec, Canada, April 1995.

[2] F. Arman and J. Aggarwal. Model-based object recognition in dense-range images - a review. *ACM Computing Surveys*, 25(1):5–43, apr 1993.

[3] A. H. Barr. Superquadrics and angle preserving transformations. *IEEE Computer Graphics and Applications*, 1(1):11–23, Jan. 1981.

[4] A. S. Bedekar and R. M. Haralick. Finding corresponding points based on bayesian triangulation. In *Proceedings, Conference on Computer Vision and Pattern Recognition* [21], pages 61–66.

[5] R. Bergevin, D. Laurendeau, and D. Poussart. Estimating the 3d rigid transformation between two range views of a complex object. In *Proceedings of the 11th International Conference on Pattern Recognition (IAPR)*, The Hague, The Netherlands, August 30-September 3 1992.

[6] P. Besl and R. Jain. Three-dimensional object recognition. *ACM Computing Surveys*, 17(1):75–145, Mar. 1895.

[7] T. Binford. Visual perception by computer. In *Conference on Systems and Control*, Miami, Florida, 1971.

[8] R. M. Bolle and D. B. Cooper. On optimally combining pieces of information, with application to estimating 3-d complex-object position from range data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(5):619–638, 1986.

[9] T. Boult and A. Gross. Recovery of superquadrics from depth information. In *Proceedings of the AAAI workshop on spatial reasoning and multisensor integration*, pages 128–137. American Association for Artificial Intelligence, 1987.

[10] K. Bowyer. Why aspect graphs are not (yet) practical for computer vision. In *IEEE Workshop on Directions in Automated CAD-Based vision*, pages 98–104, Maui, Hawaii, USA, June 1991. IEEE, New Yours, NY, USA.

[11] G. Box and G. Tao. *Bayesian inference in statistical analysis.* Addison-Wesley, 1973.

[12] J. S. Bridle. Probabilistic interpretation of feedforward classification networks outputs, with relationship to statistical pattern recognition. In F. Fogelman-Soulié and J. Hérault, editors, *NATO ASI*, volume F68. Springer-Verlag, Berlin, 1990.

[13] F. G. Callari and F. P. Ferrie. Active recognition: Using uncertainty to reduce ambiguity. In *Proceedings of the 13th International Conference on Pattern Recognition*, Vienna, Austria, Aug.25–30 1996.

[14] F. G. Callari and F. P. Ferrie. Autonomous recognition: Driven by ambiguity. In *Proceedings, Conference on Computer Vision and Pattern Recognition* [21].

[15] F. G. Callari, G. Soucy, and F. P. Ferrie. Landmark identification and pose determination in unstructured environments. Technical Report TR-CIM-96-08, Center for Intelligent Machines, McGill University, Montréal, Québec, Canada, 1996. Available via ftp at `ftp.cim.mcgill.ca` in `/pub/3d/papers/tr-cim-96-08.ps.gz`.

[16] O. Camps and R. Haralick. PREMIO: an overview. In *IEEE Workshop on Directions in Automated CAD-Based Vision*, pages 11–21, 1991.

[17] V. Chandrasekaran and T. M. Palaniswami, M. andCaelli. Spatio-temporal feature maps using gated neuronal architecture. *IEEE Transactions on Neural Networks*, 6(5):1119–1131, Sept. 1995.

[18] D. Chelberg and J. H. Yi. Range image segmentation using regularization. In *SPIE*, volume 1607, pages 336–347. Int.l Soc. for Optical Engineering, Bellingham, WA, USA.

[19] Y. Cheng and G. Medioni. Object modeling by registration of multiple range images. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pages 2724–2729, 1991.

[20] R. T. Chin and C. R. Dyer. Model-based recognition in robot vision. *Computing Surveys*, 18(1):67–108, mar 1986.

[21] Computer Society of the IEEE. *Proceedings, Conference on Computer Vision and Pattern Recognition*, San Francisco, June 1996. IEEE Computer Society Press.

[22] Describing and R. D. O. U. S. Properties. *Fan, T.J.* Springer–Verlag, New York, NY, USA, 1990.

[23] M. do Carmo. *Differential Geometry of Curves and Surfaces.* Prentice-Hall, Inc., Englewood Cliffs,New Jersey, 1976.

[24] R. O. Duda and P. E. Hart. *Pattern classification and scene analysis.* Wiley, 1973.

[25] D. Eggert and K. Bowyer. Computing the orthographic projection aspect graph of solids of revolution. In *PROC. of IEEE Workshop on the Interpretation of 3-D Scenes*, pages 102–108, Austin, Texas, Novem ber 27-29 1989. IEEE.

[26] V. Fedorov. *Theory of optimal experiments.* Academic Press, New York, 1972.

[27] F. P. Ferrie, J. Lagarde, and P. Whaite. Darboux frames, snakes, and super-quadrics: Geometry from the bottom up. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(8):771–784, Aug. 1993.

[28] P. J. Flynn and A. K. Jain. BONSAI: 3-D object recognition using constrained search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):1066–1075, Oct. 1991.

[29] S. Ghosal and R. Mehrotra. Segmentation of range images: an orthogonal moment-based integrated approach. *IEEE Transactions on Robotics and Automation*, 9:385–99, Aug. 1993.

[30] W. Grimson and T. Lozano-Perez. Localizing overlapping parts by searching the interpretive tree. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(4):469–482, 1987.

[31] S. F. Gull. Developments in maximum entropy data analysis. In J. Skilling, editor, *Maximum entropy and Bayesian methods*, pages 53–71. Kluwer Academic Publishers, 1989.

[32] A. Gupta and R. Bajcsy. Volumetric segmentation of range images of 3d objects using superquadric models. *Computer Vision, Graphics, and Image Processing*, 58:302–26, Nov. 1993.

[33] A. Gupta and R. Bajcsy. Segmentation of range images as the search for geometric parametric models. *International Journal of Computer Vision*, 14:253–77, April 1995.

[34] R. M. Haralick et al. Pose estimation from corresponding point data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-19(6):1426–1446, Nov. 1989.

[35] A. Hoover, G. Jean-Baptiste, X. Jiang, P. J. Flynn, H. Bunke, D. Goldgof, and K. Bowyer. A comparison of range image segmenmtation algorithms. 1995.

[36] B. Horn. Extended gaussians images. *Proc. IEEE*, 72(12):1671–1686, Dec. 1984.

[37] B. Horn. *Robot Vision.* MIT Press, Cambridge, Massachusetts, 1986.

[38] C. Jackins and S. Tanimoto. Octrees and their use in representing three-dimensional objects. *Computer Graphics and Image Processing*, 14(3):249–270, Nov. 1980.

[39] A. Jain and R. Hoffman. Evidence-based recogntion of 3d objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(6):783–802, June 1988.

[40] S. Kang and K. Ikleuchi. Determining 3d object pose using the complex extended Gaussian image. In *Proceedings, Conference on Computer Vision and Pattern Recognition*, pages 580–585, Lahaina, Maui, Hawaii, June 1991. Computer Society of the IEEE, IEEE Computer Society Press.

[41] S. Kaveti, E. Khwang, and H. Wang. On 3-d image segmentation based on planar and quadric approximations. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, volume 1, pages 349–353. IEEE Computer Society, IEEE, Piscataway, NJ, USA, 1994.

[42] B. B. Kimia, A. R. Tannenbaum, and S. W. Zucker. Shapes, shocks, and deformations, I: The components of shape and the reaction-diffusion space. Technical Report LEMS 105, LEMS, Brown University, May 1992.

[43] J. Koenderink. The singularities of the visual mapping. *Biological Cybernetics*, 24(1):51–59, 1976.

[44] J. Koenderink. Internal representation of solid shape with respect to vision. *Biological Cybernetics*, 32(4):211–216, 1979.

[45] M. Korn and C. Dyer. 3–D multiview object representations for model-based object recognition. *Pattern Recognition*, 20(1):91–103, 1987.

[46] D. Kriegman and J. Ponce. Computing exact aspect graphs of curved objects: Solids of revolution. In *PROC. of IEEE Workshop on the Interpretation of 3-D Scenes*, pages 116–122, Austin, Texas, November 27-29 1989. IEEE.

[47] T. Kuni, T. Satoh, and K. Yamaguchi. Generation of topological boundary representation from octree encoding. *IEEE Comp. Graph. Appl.*, 5(3):29–38, Mar. 1985.

[48] A. Lejeune and F. Ferrie. Partitioning range images using curvature and scale. In *PROC. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 800–801, New York City, New York, June 15-17 1993.

[49] A. Lejeune and F. Ferrie. Finding the parts of objects in range images. *Computer Vision, Graphics, and Image Processing*, 1995. to appear.

[50] D. Lowe, editor. *Perceptual Organization and Visual Recognition.* Kluwer, Boston, 1985.

[51] D. J. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, 1991.
[52] D. J. MacKay. The evidence framework applied to classification networks. *Neural Computation*, 4(5):698–714, 1991.
[53] D. J. MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4(4):589–603, 1991.
[54] D. J. MacKay. A practical Bayesian framework for backprop networks. *Neural Computation*, 4(3):448–472, 1991.
[55] D. J. MacKay. *Bayesian methods for adaptive models*. PhD thesis, California Institute of Technology, Pasadena, CA, USA, 1992.
[56] Madsen. Viewpoint variation in the noise sensitivity of pose estimation. In *Proceedings, Conference on Computer Vision and Pattern Recognition* [21], pages 41–46.
[57] J. Mundy and A. Zisserman, editors. *Geometric Invariance in Computer Vision*. MIT Press, Boston, 1992.
[58] M. P. A new strategy for octree representation of three dimensional objects. In *Proceedings, Conference on Computer Vision and Pattern Recognition*, Ann Arbor,Michigan,USA, June 1988. Computer Society of the IEEE, IEEE Computer Society Press.
[59] A. Pentland. Fractal-based description of natural scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):661–674, 1984.
[60] A. Pentland. Recognition by parts. In *Proceedings, 1ST International Conference on Computer Vision*, pages 612–620, London,UK, June 1987. Computer Society of the IEEE, IEEE Computer Society Press.
[61] W. Planting and C. Dyer. An algorithm for constructing an aspect graph. In *Proc. IEEE Symposium on the Foundations of Computer Science*, pages 123–131, New York, 1986. IEEE.
[62] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317(26):314–319, Sept. 1985.
[63] E. Polack. *Computational methods in optimization*. Academic Press, New York, 1971.
[64] J. Ponce and M. Brady. Toward a surface primal sketch. In *ICRA*, pages 420–425, St-Louis, MO, Mar. 1985. IEEE Computer Society, IEEE Computer Society Press.
[65] J. Ponce and O. Faugeras. An object centered hierarchical representation for 3–D objects: The prism tree. *Computer Vision, Graphics, and Image Processing*, 38(1):1–27, Apr. 1987.
[66] M. J. D. Powell. Restart procedures for the conjugate gradient method. *Mathematical Programming*, 12:241–254, 1968.
[67] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numeric Recipes in C – The Art of Scientific Computing*. Cambridge University Press, Cambridge, U.K., 1988.
[68] W. Richards and D. Hoffman. Codon constraints on closed 2d shapes. *Computer Vision, Graphics, and Image Processing*, 31:265–281, 1985.
[69] M. Rioux. Laser range finder based on synchronized scanners. *Applied Optics*, 23(21):3837–3844, Nov. 1984.
[70] B. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge, England, 1996.
[71] G. Seetharaman and C.-H. Chu. Hierarchical fusion of geometric constraints for image segmentation. In *Proceedings of the SPIE*, volume 1383, pages 582–588. SPIE.
[72] I. Shimshoni and J. Ponce. Probabilistic 3D object recogition. In *Proceedings, 5TH International Conference on Computer Vision*, pages 488–493, Cambridge, Massachusetts, June18-21 1995. Computer Society of the IEEE, IEEE Computer Society Press.
[73] A. F. M. Smith and D. J. Spiegelharter. Bayes factors and choice criteria for linear models. *Journal of the Royal Statistical Society B*, 42(2):213–220, 1980.
[74] F. Solina and R. Bajcsy. Recovery of parametric models from range images: The case for superquadrics with global deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(2):131–147, Feb. 1990.
[75] G. Soucy and F. Ferrie. Motion and surface recovery using curvature and motion consistency. *Computer Vision, Graphics, and Image Processing*, 1995.
[76] D. J. Spiegelharter and S. L. Lauritzen. Sequential updating of conditional probabilities on directed graphical structures. *NETWORKS*, 20:579–605, 1990.

[77] R. Szeliski. *Bayesian Modeling of Uncertainty in Low–Level Vision.* Kluwer Academic Publishers, Boston, MA, USA., 1989.

[78] A. Tarantola. *Inverse Problem Theory: Methods for Data Fitting and Model Parameter Estimation.* Elsevier Science Publishing Company Inc., 52, Vanderbuilt Avenue, NewYork, NY 10017, U.S.A., 1987.

[79] R. Taylor, M. Savini, and A. Reeves. Fast segmentation of range imagery into planar regions. In *Computer Vision, Graphics, and Image Processing*, volume 45, pages 42–60, 1989.

[80] N. Tishby, E. Levin, and S. Solla. Consistent inference of probabilities in layered networks. In *Proc. Internat. Joint Conf. on Neural Networks*, Washington, 1989.

[81] E. Trucco and R. B. Fisher. Experiments in curvature-based segmentation of range data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):177–182, Feb 1995.

[82] P. Whaite and F. P. Ferrie. From uncertainty to visual exploration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):1038–1049, Oct. 1991.

[83] P. Whaite and F. P. Ferrie. Active exploration: Knowing when we're wrong. In *PROC. Fourth International Conference on Computer Vision*, pages 41–48, Berlin, Germany, May 11-14 1993. Computer Society of the IEEE, IEEE Computer Society Press. See also [85].

[84] P. Whaite and F. P. Ferrie. Autonomous exploration: Driven by uncertainty. Technical Report TR-CIM-93-17, Center for Intelligent Machines, McGill University, Montréal, Québec, Canada, 1993. *Accepted for publication in PAMI, October 1995.* Available via ftp at `ftp.cim.mcgill.ca` in `/pub/3d/papers/tr-cim-93-17.ps.gz`.

[85] P. Whaite and F. P. Ferrie. On the sequential determination of model misfit. Technical Report TR-CIM-94-11, Center for Intelligent Machines, McGill University, Montréal, Québec, Canada, 1993. *Accepted for publication in PAMI, October 1995.* Available via ftp at `ftp.cim.mcgill.ca` in `/pub/3d/papers/tr-cim-93-17.ps.gz`.

[86] P. Whaite and F. P. Ferrie. Autonomous exploration: Driven by uncertainty. In *Proceedings, Conference on Computer Vision and Pattern Recognition*, pages 339–346, Seattle, Washington, June 21-23 1994. Computer Society of the IEEE, IEEE Computer Society Press. See also [84].

[87] M. D. Wheeler and K. Ikeuchi. Sensor modeling, probabilistic hypothesis generation, and robust localization for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(3):252–265, Mar. 1995.

[88] A. P. Witkin. Scale-space filtering. In *JCAI*, pages 1019–1021, 1983.

[89] N. Yokoya and M. D. Levine. Range image segmentation based on differential geometry: A hybrid approach. Technical Report McRCIM-TR-CIM 87-16, Computer Vision and Robotics Laboratory, McGill Research Center for Intelligent Machines, McGill University, Québec, Canada, Sept. 1987.

Centre for Intelligent Machines, McGill University, 3480 University St., Montreál, Que., Canada, H3A 2A7

E-mail address: {`franco,ferrie,soucy`}`@cim.mcgill.ca`