Autonomous Exploration: Driven by Uncertainty

Peter Whaite and Frank P. Ferrie, *Member, IEEE*

Abstract—Passively accepting measurements of the world is not enough, as the data we obtain is always incomplete, and the inferences made from it uncertain to a degree which is often unacceptable. If we are to build machines that operate autonomously, they will always be faced with this dilemma, and can only be successful if they play a much more active role. This paper presents such a machine. It deliberately seeks out those parts of the world which maximize the fidelity of its internal representations, and keeps searching until those representations are acceptable. We call this paradigm *autonomous exploration*, and the machine an autonomous explorer.

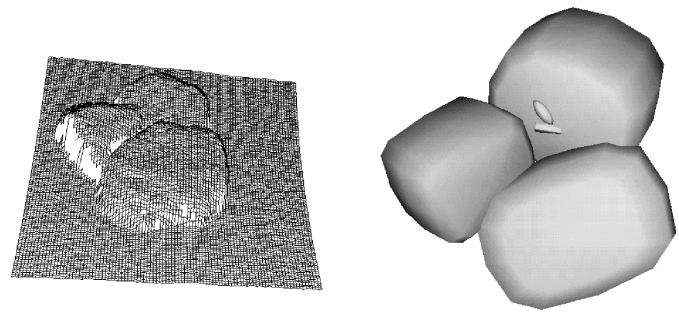
This paper has two major contributions. The first is a theory that tells us how to explore, and which confirms the intuitive ideas we have put forward previously. The second is an implementation of that theory. In our laboratory, we have constructed a working autonomous explorer and here, for the first time, show it in action. The system is entirely bottom-up and does not depend on any a priori knowledge of the environment. To our knowledge, it is the first to have successfully closed the loop between gaze planning and the inference of complex 3D models.

Index Terms—Autonomous exploration, active vision, visual servoing, artificial perception, unstructured environments, volumetric models, superellipsoids, next best view, theory of optimal experiments.

1 Introduction

One can define autonomous exploration as a process in which an observer can interact with its surroundings by moving about and collecting information in order to learn about its environment. This ability is essential for autonomous systems which must operate in unstructured environments where it is difficult (if not impossible) to characterize the environment beforehand. Consider, for example, a mobile robot designed to collect rock samples for planetary exploration [1]. In order to grasp and manipulate such samples, information is required about their three-dimensional shape. But given the wide range of shapes that are possible, it is not feasible to represent each and every instance. Shape descriptions must be computed from more general purpose models that can be adapted according to measurements obtained by sensors. In the context of artificial perception, the latter often takes the form of determining the parameters of some model used to reflect the salient properties of the environment [2], [3].

Fig. 1a shows a range map of a rock pile obtained with a laser range-finding system. From an analysis of the geometric structure of the acquired surfaces, the data are partitioned into patches corresponding to the component rocks. An approximation of the position, orientation, and shape of each component is then determined by fits to superquadric models (Fig. 1b) [4]. While this strategy appears to work well in the example shown, it is in fact flawed. This can be seen in the example in Fig. 2 which shows the same fitting process applied to points sampled from the surface of a noisy hemisphere. Each of the resulting models (Fig. 2b) describes the data to within the same error of fit [5], [6], yet the models look quite different as they move away from the data. The problem is that the data acquired do not sufficiently constrain the model. This should not be surprising given that only part of the surface is visible in a given view. The example of Fig. 1 worked because an additional constraint was available, namely the distance from the camera to the supporting plane.



(a) (b)

Fig. 1. (a) Laser rangefinder image of a rock pile. (b) Model of the rock pile using superquadrics.

Without additional information in the form of such constraints, there is no alternative but to collect it in the form of additional data. In this respect, some data are better than others, so the system must actively seek out those places in the world that have the most useful information. As stated by Bajcsy [7] and by Bajcsy and Campos [8], the system needs to *explore* its environment and it must keep doing so until there is a sufficient basis from which to make useful inferences.

• *The authors are with the Artificial Perception Laboratory, Centre for Intelligent Machines, McGill University, 3480 University, Montréal, Québec, Canada H3A 2A7. E-mail: {ferrie, peta}@cim.mcgill.ca.*

Manuscript received Mar. 14, 1994. Recommended for acceptance by J. Aloimonos. For information on obtaining reprints of this article, please send e-mail to: transpami@computer.org, and reference IEEECS Log Number P96112.

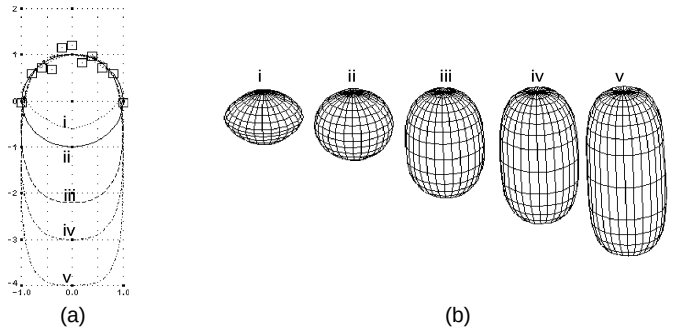


Fig. 2. Model uncertainty. (a) Data points sampled from the surface of a noisy hemisphere. (b) Fits to the data using superellipsoid models. Each of the five models shown fits the data to within the same tolerance.

The idea of moving a sensor to constrain interpretation has a long history in the computer vision literature. Many of the methods rely on global geometric arguments to overcome the limitations of the sensor in specific situations. For example, Connolly [9], then Ahuja and Veenstra [10] have considered the problem of the views needed to build an octree representation of a 3D scene; Tarabanis and Tsai [11] have worked on a theoretical analysis of the best camera viewpoint for detecting a generic feature; and, more recently, Maver and Bajcsy [12] developed a technique for filling in the range image shadows when sampling a scene with a light stripe range finder.

Some important general principals were established with the introduction of the active vision paradigm by Aloimonos and Bandyopadhyay[13]. The basis of the approach is a parametrized model of how the sensor views features in the scene. Continuously varying the view parameters causes the observed features to undergo measurable local transformations which can be used to simplify and constrain the computation of unknown scene parameters. For example, Kutulakos and Dyer [14] cleverly exploit the differential properties of smooth surfaces to model local changes in the appearance of an occluding contour due to camera movement. This knowledge shows them how to position the camera, first to extract occluding contours from an edge map, and then to use the extracted contours to sweep out the complete 3D shape.

Much of the power of the active vision paradigm comes from the simplification of only having to model local changes to local features, so it is more applicable to problems in low and intermediate vision where local features play the major role. However, in a complex vision system, these problems are only the first steps towards a description of the world which will enable the system to perform some specific task, so if there are any deficiencies in the low levels it becomes essential that they be taken into account. For example, it is impossible to reconstruct some surface concavities using the method of occluding contours proposed by Kutulakos and Dyer. Without that knowledge, decisions based on the reconstructed surface could result in a system failure. With it, other instrumentation can be called upon to make up for the method's shortcomings.

It has also been demonstrated that "high-level" control of data acquisition is very useful. For example, Birnbaum et al. [15] present a system (BUSTER) which uses a generative causal semantics to control the visual exploration of arbitrary stacked block structures. The system works by encoding the knowledge that the scene is stable under the force of gravity in a simple set of rules. The rules are used to direct visual attention to search for blocks which make an unstable scene stable.

Thus, in complex systems such as the one we are working on, the information used to select sensor viewpoints can come from many different sources and must be combined in ways that go beyond the principals established by the active vision paradigm. One can attempt to formulate general frameworks to represent and solve these problems [16] but tractable solutions often require the development of specific insight into the structure of the problem. We attempt to avoid this dilemma with an active vision approach that generalizes the characterization and manipulation of uncertainty. As a consequence, it can be applied wherever the interaction between a sensor and its environment is modeled parametrically.

We begin development of the theory in Section 2 with a description of a system where the location of a sensor is determined by a set of control parameters, and in which the interaction between the sensor and its environment is modeled by the linear combination of an arbitrary set of basis functions. We show how to find a maximum likelihood estimate of unknown model parameters from a set of noisy measurements, and illustrate that the parameter covariances represent and encapsulate the model uncertainty.

In Section 3, we continue with an information based measure of uncertainty, the determinant of the covariances, and consider the problem of how to reduce it incrementally by taking a single extra measurement. We find a theoretical solution to this problem which is identical to a proposition we have made in previous work [6]—that the best sensor locations are those where our ability to predict is worst.

This leads to a gaze-planning strategy, described in Section 4, that uses model uncertainty as a basis for selecting viewpoints. We show theoretically that the strategy ensures convergence when applied to linear models, and present experimental results for nonlinear superellipsoid models which verify that the linear theory can be applied as a local approximation.

By closing the loop around bottom-up vision with this gaze-planning strategy, we design in Section 5 an exploration system that is capable of autonomously building a description of its environment from a sequence of exploratory probes. This leads to the implementation in Section 6, and a sequence of experimental results which show how the system performs on real scenes obtained with a mobile laser rangefinding system.

2 Linear Model Inference

Although the volumetric models we use to represent surfaces in a 3D scene are highly nonlinear, many of the basic concepts and insights are obtained from a study of the linear case. Nonlinear analytic solutions are usually hard to obtain and invariably one must resort to numerical iterative techniques to get results. Once this is done, however, the

system can be linearized around the solution, and the linear analysis applied. Provided that perturbations in the state of the system are small enough, the linear analysis is valid and not as severe a restriction as might first appear.

2.1 The Linear Model

Consider the scenario where we have a sensor making measurements of a physical system and where the location of the sensor is determined by specifying a vector of control parameters $\mathbf{x}$.[1] In a linear model, data measurements can be predicted by a linear combination of basis functions defined over the space of control parameters. That is given known model parameters $\mathbf{m}$ the measurement obtained at location $\mathbf{x}_i$ can be written in the form $d_i = \mathbf{g}_i^T\mathbf{m}$ where $\mathbf{g}_i^T = (g_1(\mathbf{x}_i), \ldots, g_p(\mathbf{x}_i))$ are the basis functions evaluated at $\mathbf{x}_i$.

The basis functions themselves do not have to be linear in the control parameters. For example, suppose our sensor is a depth probe constrained to move in a horizontal plane. Its location is given by the Cartesian coordinates $\mathbf{x}_i^T = (x_i, y_i)$, and the measurement d_i it makes there is the vertical distance to some surface. We can choose to model the surface as a general quadratic $d_i = ax_i^2 + bx_iy_i + cy_i^2 + dx_i + ey_i + f$, then the model parameters are $\mathbf{m}^T = (a, b, c, d, e, f)$, and the basis functions $\mathbf{g}_i^T = (x_i^2, x_iy_i, y_i^2, x_i, y_i, 1)$ are nonlinear in the control parameters.

2.2 Uncertainty and the Maximum Likelihood Solution

In general, the model parameters are not known, and the whole purpose of making measurements is to solve the *inverse problem*, that is to find model parameters that explain the observed measurements. If we have made n measurements such a solution amounts to solving the linear system of equations $\{d_i = \mathbf{g}_i^T\mathbf{m}\,;\, i = 1 \ldots n\}$ which can be written in the form $\mathbf{d} = \mathbf{G}\mathbf{m}$ where the $\mathbf{g}_i$ form the rows of $\mathbf{G}$. When the data is contaminated with noise, it is not possible to find an exact solution, but for measurement errors randomly and independently sampled from a normal distribution with zero mean and variance σ^2, it can be shown that the *maximum likelihood estimate* $\hat{\mathbf{m}}$ of the true model $\mathbf{m}_T$ is given by the pseudo inverse

$$\hat{\mathbf{m}} = \left(\mathbf{G}^T\mathbf{G}\right)^{-1}\mathbf{G}^T\mathbf{d}. \tag{1}$$

Furthermore, the parameter errors $\hat{\mathbf{e}}(\mathbf{m}_T) = \mathbf{m}_T - \hat{\mathbf{m}}$ are distributed as a zero mean p-variate normal distribution with covariances

$$\mathbf{C} = \sigma^2\left(\mathbf{G}^T\mathbf{G}\right)^{-1} = \sigma^2\mathbf{H}. \tag{2}$$

The presence of random errors in the inverse solution is an indication of its inherent uncertainty or nonuniqueness. Consider the quadratic form

$$Q(\mathbf{m}) = \hat{\mathbf{e}}(\mathbf{m})^T\mathbf{C}^{-1}\hat{\mathbf{e}}(\mathbf{m}) = \frac{1}{\sigma^2}\hat{\mathbf{e}}(\mathbf{m})^T\mathbf{H}\hat{\mathbf{e}}(\mathbf{m}) \tag{3}$$

that defines the shape of the normal parameter error distribution. Now $Q(\mathbf{m}) = Q(\mathbf{m}_T)$ defines a hyper-surface on which the true model must lie. Because $\mathbf{C}^{-1}$ is symmetric and positive definite that surface is ellipsoidal.

We do not know the position of this surface because we do not know the value of $\mathbf{m}_T$, but it is a well-known result of statistical theory that $Q(\mathbf{m}_T)$ is randomly sampled from a chi-square distribution with p degrees of freedom [17]. For some *confidence level* γ, we can find from that distribution a number χ_γ^2 for which there is a probability of γ that $Q(\mathbf{m}_T) < \chi_\gamma^2$. It follows that there is also a probability of γ that the ellipsoid

$$\sigma^2 Q(\mathbf{m}) = \hat{\mathbf{e}}(\mathbf{m})^T\mathbf{H}\,\hat{\mathbf{e}}(\mathbf{m}) = \sigma^2\chi_\gamma^2 \tag{4}$$

will enclose the true model,[2] and for this reason it is called the *ellipsoid of confidence*. The ellipsoid of confidence gives us a useful visual image of the nonuniqueness of the inverse solution as it shows us the region of model parameter space in which any of the models could be the true one.

3 Reducing Uncertainty

The parameter covariances can be used to communicate the uncertainty to the tasks which make use of the inverse solution. By applying classic statistical methods, the solution can be tested to see if it meets standards of acceptability predetermined to keep system failure rates below bearable levels. When the uncertainty is such that it does not, we must take steps to improve the uniqueness until it does.

One way is to build better sensors with lower noise figures. It can be seen from (4) that lower values of σ^2 simply scale the ellipsoid of confidence to be smaller. A better way is to affect the character of the uncertainty through the choice of measurement locations. $\mathbf{H}$ is dependent only upon the $g_j(\mathbf{x}_i)$, that is on the form of the basis functions and the locations of the measurements.[3] As $\mathbf{H}$ defines the eccentricity, size, and pose of the ellipsoid of confidence we have a potentially powerful method for controlling uncertainty. By simply selecting appropriate $\mathbf{x}_i$, we can cause the ellipsoid of confidence to fall in regions of parameter space which meet criterion of acceptability imposed by the task at hand.

3.1 A Measure of Uncertainty

A central question is what constitutes an appropriate criterion of acceptability. Strictly, this question can only be answered in an operational context, but a generally useful criteria is one based on the well-known Shannon entropy, and which measures the amount of information contained in the probability distribution representing the parameter errors.[4] It

1. By "location," we mean the sensor's location in the space of control parameters. This could be its physical location, but could also be many other things, e.g., the direction of gaze, sampling density, beam intensity, etc.

2. This statement is often misinterpreted to mean there is a probability of γ that $\mathbf{m}_T$ is inside *this particular* ellipsoid. This is not so—it means that when (4) is repeatedly computed from many independently sampled data sets, then $\mathbf{m}_T$ will fall inside the computed ellipsoids γ% of the time.

3. It is perhaps surprising that the uncertainty is totally independent of the true model and, therefore, of the actual measurements. However, this is not true when the model is nonlinear.

4. Recently, we have shown that this criterion is of fundamental importance in recognition context.

can be shown that maximizing this information is equivalent to a minimization of the determinant of the parameter covariances [18]. Geometrically, det ($\mathbf{C}$) is proportional to the square of the *volume* of the ellipsoid of confidence. Small values correspond to small volumes of model parameter space, which indicate that the true parameters are well localized, and that the knowledge or information we have about them is high.

3.2 The Location of Uncertainty

Here, we will concentrate on what we call the *incremental problem*:

Given covariances $\mathbf{C}_n$ *computed from n measurements, what single additional sensor location* $\mathbf{x}_{n+1}$ *will minimize* $\det(\mathbf{C}_{n+1})$*?*

We note that this is equivalent to maximizing $\det(\mathbf{H}_{n+1})$. Each additional measurement incrementally updates $\mathbf{H}$, so after $n+1$ measurements, its value is

$$\mathbf{H}_{n+1} = \mathbf{H}_n + \mathbf{g}_{n+1}\mathbf{g}_{n+1}^T, \tag{5}$$

where $\mathbf{g}_{n+1}$ are the basis functions evaluated at each new location $\mathbf{x}_{n+1}$. After factoring $\mathbf{H}_n$ out on the right, we get that the determinant is

$$\det(\mathbf{H}_{n+1}) = \det\left(\mathbf{I} + \mathbf{g}_{n+1}\mathbf{g}_{n+1}^T\mathbf{H}_n^{-1}\right)\det(\mathbf{H}_n). \tag{6}$$

Further simplification requires us to compute the determinant of the quantity $\mathbf{I} + \mathbf{g}_{n+1}\mathbf{g}_{n+1}^T\mathbf{H}_n^{-1}$. Because $\mathbf{g}_{n+1}\mathbf{g}_{n+1}^T$ is of rank 1, it follows that there is only one nonunit eigenvalue of $\mathbf{I} + \mathbf{g}_{n+1}\mathbf{g}_{n+1}^T\mathbf{H}_n^{-1}$ and that its value is $1 + \mathbf{g}_{n+1}^T\mathbf{H}_n^{-1}\mathbf{g}_{n+1}$. As the determinant of a matrix is the product of the eigenvalues (6) simplifies to

$$\det(\mathbf{C}_{n+1}) = \det(\mathbf{C}_n)\Big/\left(1 + \frac{\mathbf{g}_{n+1}^T\mathbf{C}_n\mathbf{g}_{n+1}}{\sigma^2}\right). \tag{7}$$

There is an important interpretation which can be placed on the quantity $\mathbf{g}_{n+1}^T\mathbf{C}_n\mathbf{g}_{n+1}$ and which confirms some of our intuitive notions as to where the best sensor placement is. The model parameters $\hat{\mathbf{m}}_n$ estimated from the first n measurements allow us to *predict* the measurement that will be obtained at location $\mathbf{x}_{n+1}$ according to $\hat{\mathbf{d}}(\mathbf{x}_{n+1}) = \mathbf{g}_{n+1}^T\hat{\mathbf{m}}_n$. However, there are random errors in the estimated model parameters so we would expect there to be random errors in the predicted measurement as well. It can easily be shown that the variance of $\hat{\mathbf{d}}$ is

$$\sigma_D^2(\mathbf{x}_{n+1}) = \mathbf{g}_{n+1}^T\mathbf{C}_n\mathbf{g}_{n+1}. \tag{8}$$

This is called the *prediction variance* at location $\mathbf{x}_{n+1}$.

The ratio of det ($\mathbf{C}$) before and after an additional measurement is seen from (7) to be

$$\frac{\det(\mathbf{C}_{n+1})}{\det(\mathbf{C}_n)} = \frac{1}{1 + \sigma_D^2(\mathbf{x}_{n+1})/\sigma^2}. \tag{9}$$

It shows us

1) that adding any data will always result in a reduction of det ($\mathbf{C}_n$), and
2) that det ($\mathbf{C}_{n+1}$) can be minimized by taking a measurement at the location when σ_D^2 is largest.

As our intuition might lead us to expect, any additional data is beneficial but the best locations to gather new measurements are those where our ability to predict is worst. A similar result can also be found from Fedorov in his theory of optimal experiments [19].

4 Looking: The Gaze Planning Strategy

The theory tells us the best place to take a *single measurement* but it is rarely the case that this measurement will meet our needs. Instead, we have to collect data at a sequence of locations $\mathbf{x}_1, \mathbf{x}_2, \cdots \mathbf{x}_n$, until the estimated parameter covariances $\mathbf{C}_n$ are acceptable. Here, we consider the problem of how to choose such a sequence. We will refer to a sequence of sensor locations as a "gaze trajectory," and to the problem of choosing a trajectory as a "gaze planning strategy." The approach we have taken is inspired by and presented in the context of 3D volumetric modeling. However, the methodology is applicable to a much wider range of problems.

There are many trajectories that will eventually result in an acceptable value of $\mathbf{C}_n$. Ideally, the correct trajectory is that which is optimal in the operational context but because this would often require task specific knowledge to be embedded within the planning strategy, we would prefer to use a more generally useful criteria, even if it is at the expense of some operational optimality.

We start by considering linear models and give, in Section 4.1, the theoretical guarantee that a strategy which always moves towards uncertain viewpoints has the important property that the determinant of the covariances will converge below any arbitrary value.

In Section 4.2, we show how the linear theory can be applied in the more interesting nonlinear case. It is not possible to compute nonlinear gaze planning strategies off-line so in Section 4.3, we develop a general iterative gradient strategy based on the model estimate at each iteration. We then present in Section 4.4 a specific implementation of this strategy, and the kinds of trajectories obtained when exploring the surfaces of superellipsoidal objects. Finally, in Section 4.5, we consider additional problems encountered when scenes are explored in the real world and outline the way in which we can decrease their effect. Ultimately, however, such problems are beyond the domain of the strategy, and point to the necessity of an overseer with task specific knowledge.

4.1 The Convergence of Linear Models

When the model is linear we see from (9) that an additional measurement always results in some reduction of det ($\mathbf{C}_n$) irrespective of the sensor's location, and, therefore, that any gaze trajectory must result in a monotonically decreasing sequence det ($\mathbf{C}_1$), det ($\mathbf{C}_2$),.... However, for some trajectories, the sequence can converge to a positive, nonzero value, and it may prove impossible to reduce det ($\mathbf{C}_n$) to an adequate level. Fortunately, as the following argument shows, our condition that we always measure at locations of high σ_D^2 prevents this from happening.

First, we note that $\mathbf{C}_n = \sigma^2\mathbf{H}_n^{-1}$ so $\det(\mathbf{C}_n) \to 0$ provided $\det(\mathbf{H}_n) \to \infty$ as $n \to \infty$. We will not go into the mathematical details here, but it can be shown that the sensor trajectories for which $\det(\mathbf{H}_n)$ does not diverge are those where $\mathbf{g}(\mathbf{x}_n) \to \mathbf{0}$ [20]. As an illustration, consider the problem of estimating from depth probes z the slant α and tilt β of a planar surface known to pass through the origin. We can model this with the equation $z = \alpha x + \beta y$, where the sensor is located at $\mathbf{x}^T = (x, y)$, the basis functions are $\mathbf{g}(\mathbf{x}) = \mathbf{x}$, and the model parameters are $\mathbf{m}^T = (\alpha, \beta)$. If the sensor takes a trajectory that approaches the origin, then the components of $\mathbf{H}_n$ could converge. For example, if the trajectory is along the straight line $\mathbf{x}_j^T = (x_0 r^j, y_0 r^j)$, $|r| < 1$ then

$$\mathbf{H}_\infty = \lim_{n\to\infty} \mathbf{H}_n = \mathbf{H}_0 + \frac{r^2}{1-r^2}\begin{pmatrix} x_0^2 & x_0y_0 \\ x_0y_0 & y_0^2 \end{pmatrix}, \tag{10}$$

where $\mathbf{H}_0$ is the value of $\mathbf{H}_n$ obtained before the sensor started moving towards the origin. The eigenvalues of $\mathbf{H}_\infty$ are nonzero and finite so $\det(\mathbf{C}_n)$ will *never* converge to zero in this particular case. Many, but not all, of the trajectories that spiral inward to $\mathbf{g}(\mathbf{x}) = 0$ will behave similarly.

Places where $\mathbf{g}(\mathbf{x}) = \mathbf{0}$ are rather special in that they are noninformative. First, because $\mathbf{g}\mathbf{g}^T = \mathbf{0}$ any measurements taken there do not change the value of $\mathbf{H}_n$. Second, the prediction variance $\sigma_D^2 = \mathbf{g}^T\mathbf{H}^{-1}\mathbf{g}$ is always zero no matter what the value of $\mathbf{H}$. That is, we know a priori what the model is at these places (in the example above, the surface was known to pass through the origin). There is no point in taking measurements from locations where $\mathbf{g}(\mathbf{x}) = \mathbf{0}$ because they can not contribute anything to our knowledge of the model.

Furthermore, for some local neighborhood around the place where $\sigma_D^2 = 0$ (provided, of course, that $\mathbf{g}(\mathbf{x})$ is continuous over that neighborhood), σ_D^2 increases monotonically as one moves *away* from the location where $\sigma_D^2 = 0$. A gaze planning strategy that drives the sensor towards locations of maximum σ_D^2 will avoid places where $\mathbf{g} = \mathbf{0}$, and, therefore, ensure that $\det(\mathbf{C}_n)$ converges to zero.

4.2 Nonlinear Models

Because of their descriptive power, we use nonlinear superellipsoid models to describe objects in the scene. Given that we have a theory of where to take measurements in a linear system, we would like to know if, and to what extent, that theory can be usefully applied to nonlinear models.

In our laboratory, we sample 3D surface coordinates with a laser range scanner and, after several layers of "bottom-up" processing, infer those superellipsoid models which best explain the measurements [2]. Points $\{\mathbf{s}_i, i = 1, \ldots, n\}$ on the surface of a superellipsoid model with parameters $\mathbf{m}$ satisfy the implicit equation $D(\mathbf{s}_i, \mathbf{m}) = D_i(\mathbf{m}) = 0$, which is highly nonlinear in both $\mathbf{s}_i$ and $\mathbf{m}$ [5]. In fact, D has the metric property that it is the radial distance of $\mathbf{s}_i$ from the surface. Therefore, when the sensor returns noisy measurements, such that $D(\mathbf{s}_i, \mathbf{m})$ is randomly sampled from a zero-mean normal distribution with variance σ^2, a maximum likelihood estimate of the true parameters can be found by finding those model parameters $\hat{\mathbf{m}}$ which minimize $\sum_{i=1}^{n} D_i(\mathbf{m})^2$. Because of the nonlinearities, iterative techniques must be employed.

Once the solution is obtained, the covariances are found by linearizing the model around $\hat{\mathbf{m}}$. It is easy to show that this gives a locally linear model where

$$\mathbf{g}_i = \frac{\partial D}{\partial \mathbf{m}}(\mathbf{s}_i, \hat{\mathbf{m}}), \tag{11}$$

is the Jacobian of D evaluated for $\mathbf{s}_i$ on the surface of $\hat{\mathbf{m}}$, and, therefore, by (8) that the prediction variance at a point on the surface of a superellipsoid model is $\mathbf{g}_i^T\hat{\mathbf{C}}\mathbf{g}_i$.

There is an implicit assumption in the nonlinear solution that a posteriori parameter distribution is Gaussian. In general, this is not true for two reasons:

1) the errors in $D(\mathbf{s}, \mathbf{m})$ will usually not be Gaussian, and
2) the basis functions $\mathbf{g}_i$ exhibit random variation due to the noisy measurements $\mathbf{s}_i$.

This variation combines multiplicitatively with the assumed Gaussian randomness in data errors, and results in a distribution that is not normal nor even symmetric. One consequence is that the maximum likelihood solution will, in general, be biased, so on repeated trials the expected value of $\hat{\mathbf{m}}$ will differ significantly from the true value $\mathbf{m}_T$. The seriousness of this depends upon the sensor noise level, and on the distribution of sensor locations. For superellipsoid models, the fitted surface stays within the envelope of the measurement noise, so provided the sensor noise is low, and that the data obtained constrains the model parameters, any biases due to nonlinearities will be small. During the initial stages of exploration, the parameters will not be well constrained so estimation bias could cause problems, for example, by falsely estimating the position and size of the model. However, the whole purpose of exploration is to find data that do constrain the model parameters, so ultimately this will ensure that bias is minimized.

Another consequence of a nonnormal parameter distribution is that the covariances do not fully account for the variation, and are generally an optimistic estimate of the expected parameter errors. This is problematic when Kalman filter techniques are applied to nonlinear models (the extended Kalman filter) because the weighting new data receives is based on how believable it is. Ultimately, this can result in a situation where less and less importance is attached to new data, and more and more to an estimate which diverges increasingly from the correct value [21]. The exploration strategy we propose differs fundamentally from a Kalman filter in that state updates are based only on the data collected and not at all on any previous state estimate. We do not observe the divergence exhibited by extended Kalman filters.

It is probably not wise to trust the parameter covariances if they are used probability computations, for example, when evaluating the risk of a course of action. For these

purposes, a more general probability distribution should be adopted, such as sums of Gaussians [18], or even grid-based methods [16]. For exploration, this could be a problem except, as we shall see later, the strategy we adopt follows the gradient of σ_D^2, so it is not the magnitude of the parameter covariances that is important, but their "direction." Provided we operate within the region of parameter space where the linear approximation is valid, then we can expect the gradient to be valid as well.

4.3 The Gradient Strategy—A General Nonlinear Gaze Planning Strategy

Before we can apply the linear analysis to nonlinear models, there is an important and fundamental difference which must be taken into account. When the model is linear, det ($\mathbf{C}_n$) does not depend on the model being measured, so it is always possible to find an optimal trajectory (even an operationally optimal trajectory), even if it means off-line precomputation of trajectory numerically. This is not true when the model is nonlinear as the local linear approximations for $\mathbf{g}_i$, and for $\mathbf{C}$ and σ_D^2, are all dependent upon $\hat{\mathbf{m}}$. As $\hat{\mathbf{m}}$ is estimated from the measurements taken of the model in the scene, we are placed in the untenable position of having to know the unknown model before an optimal trajectory can be computed. It is not possible to find truly optimal gaze trajectories for nonlinear models.

In view of the dependency of σ_D^2 upon $\hat{\mathbf{m}}$, we adopt the following general iterative strategy. At each step n of the gaze trajectory, we compute $(\sigma_D^2)_n = \sigma_D^2(\mathbf{x}, \hat{\mathbf{m}}_n)$ using the current estimate of the model $\hat{\mathbf{m}}_n$. The next sensor trajectory location $\mathbf{x}_{n+1}$ is chosen to be that which maximizes $(\sigma_D^2)_n$, but subject to the constraint that it lie within the region of sensor locations for which the linear approximation is valid. Once $\mathbf{x}_{n+1}$ is found, the sensor is moved there, and an additional measurement taken. The model estimate is then updated by refitting to a data set in which the new measurement is appended to all of those obtained previously. The process repeats using the updated estimate $\hat{\mathbf{m}}_{n+1}$, and runs until the parameter covariances meet some operationally defined criteria of acceptability.

A problem with the approach is that it is difficult to determine the region of sensor locations over which the nonlinear relationships of the model can be approximated sufficiently well by the linearized form (11). What we suspect, and what our observations suggest, is that the region is indicated by low values of $(\sigma_D^2)_n$, but it is not clear how to quantify the linear approximation error in σ_D^2. In fact, the size of the approximation error is not important provided the monotonic relationship between σ_D^2 and the decrease in det ($\mathbf{C}_n$) is preserved. Previous work we have done strongly suggests that this relationship remains true even when the approximation errors are large [6]. In practice, we have observed that the region of allowable sensor locations can include regions of large σ_D^2 without any noticeable degradation in performance.

However, because of worries about the validity of the linearized theory, we took a conservative approach in which the sensor was always moved in small steps. Our rationale was that in the region around the current sensor location, the position of the surface would be well-known due to the measurement already made there, so locally the difference between the true and estimated models would be small, and the linearized form would serve as an adequate approximation. Thus, instead of globally maximizing (8), we do it in the local neighborhood of the current location, and move the sensor in the direction of maximum of σ_D^2. In effect, the scanner follows an approximation of $\partial\sigma_D^2/\partial\mathbf{x}$, so we will refer to this as the *gradient strategy*.

At first glance, the gradient strategy might seem to be an implementation of the classical gradient ascent method but there is an important difference. The form $\sigma_D^2(\mathbf{x})$ being ascended is continuously changing as new data are added. When a measurement is taken from a location where $\sigma_D^2(\mathbf{x})$ is high, the prediction variance there will be reduced to the level of sensor noise. The effect of this reduction is to "push" the sensor away from the current location on the next iteration and, thus, to help overcome a major problem of gradient methods—that the sensor trajectory will falter on top of a local maximum.

4.4 The View Sphere Gradient Strategy

Here, we present an implementation in which a laser range scanner is able to orbit about a single superellipsoidal object. The location of the scanner is given by its position on the surface of a view sphere, and is directed so it always points towards the sphere's center. When describing the view sphere location, we will usually refer to its radius $\wp$, latitude ϑ, and longitude φ; but because this parametrization has singularities at the poles, a better way to represent sensor location $\mathbf{x}$ is by a 3D vector from the center of the view sphere, to the position of the sensor on the sphere's surface.

The first problem is to compute the gradient $\partial\sigma_D^2/\partial\mathbf{x}$ of the prediction variance. However, given the expression we have in (11), σ_D^2 is a function of the position $\mathbf{s}$ on the surface of the estimated model, and not of the sensor location $\mathbf{x}$. The relationship between these quantities $\mathbf{s} = \mathbf{s}(\mathbf{x}, \hat{\mathbf{m}})$ is essentially a ray tracing problem. That is, for a given scanner gaze, we need to compute where the laser beam will hit the surface on the estimated model. Provided the form of $\mathbf{s}(\mathbf{x}, \hat{\mathbf{m}})$ is known and is continuous around $\mathbf{x}$ then it is, in principal, an easy matter to obtain the gradient analytically. Unfortunately, we fail on both counts as $\mathbf{s}(\mathbf{x}, \hat{\mathbf{m}})$ is not always continuous (Fig. 3), and a closed form solution to the ray tracing problem is unknown for superellipsoid models.

Because there is no general analytic solution, we employ the numerical technique illustrated in Fig. 4. Essentially, we search for the maximum σ_D^2 on the view sphere at a fixed geodesic distance from the current scanner location. The scanner is circled around what we call the *search circle*, thus, generating a cone of laser beams directed towards the center of the view sphere. We use Newton's method to find where each beam strikes the surface of the estimated model,

and if it does we compute σ_D^2 at that spot. The direction of sensor travel is chosen to be towards the circle position which resulted in the maximum value of σ_D^2.

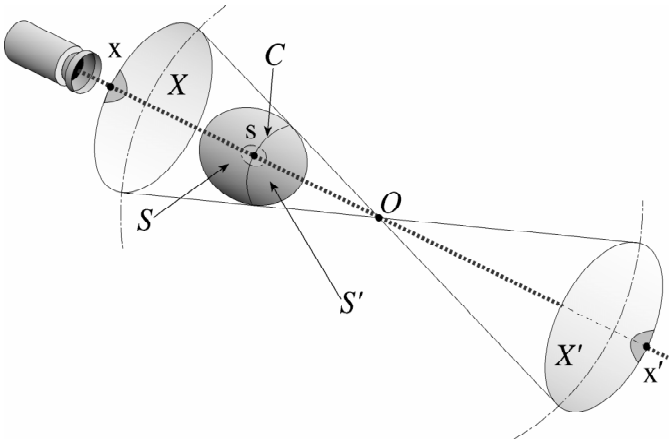


Fig. 3. The mapping between **x** and **s** is discontinuous. When the model does not enclose the view sphere center *O*, there will be a closed contour *C* where the scanner beam is tangential to the model's surface. The contour divides the model into two regions *S* and *S′* which map into the disjoint view sphere regions *X* and *X′*, respectively. Thus, the point **s** on the contour can be "seen" by the scanner when it is positioned at either **x** or **x′**, and a small movement across the contour will require a discontinuous jump in view sphere position between **x** and **x′**.

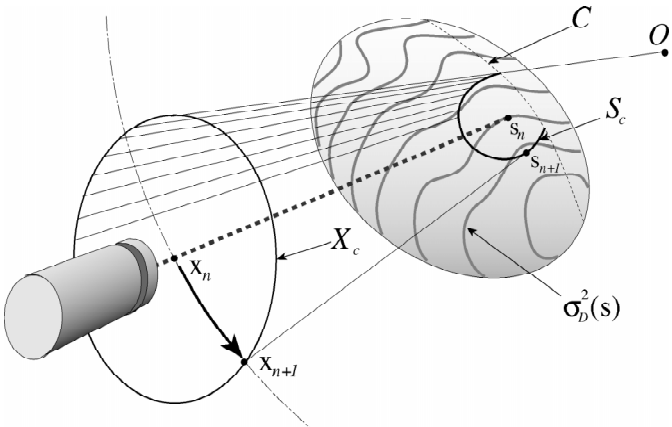


Fig. 4. The numerical solution to $\partial\sigma_D^2/\partial\mathbf{x}$. The search circle X_c on the surface of the view sphere is at a constant geodesic radius from the current scanner location $\mathbf{x}_n$, and its image S_c is formed where the cone of beams directed towards the center of the view sphere *O* intersect the surface of the model. Here, we show a case where some of the beams emanating from X_c miss the surface. The next scanner location $\mathbf{x}_{n+1}$ is the one which projects to the surface coordinate $\mathbf{s}_{n+1}$ that maximizes $\sigma_D^2(\mathbf{s})$ on S_c. The gradient $\partial\sigma_D^2/\partial\mathbf{x}$ is approximated by the direction of the arrow from $\mathbf{x}_n$ to $\mathbf{x}_{n+1}$.

We investigated the behavior of the above strategy by running simulations on various superellipsoid models positioned in the center of a view sphere. After initially fitting a model to data scanned from above (ϑ = 90°, φ = 0°), the gradient of σ_D^2 on the view sphere was evaluated numerically as shown in Fig. 4. The search circle radius θ_c was set to 20°, a value based on the proximity of the scanner to the model. In the more general situation, it could be set to adapt as the unknown model evolves. The scanner was stepped in the direction of maximum uncertainty by 20°, a noisy measurement was obtained at the new location, added to the data set, and the current estimate of the model updated using the newly enhanced data. The process was repeated for 100 iterations.

The view sphere trajectories from 40 separate explorations of an ellipse and a block are shown in Fig. 5. Because the measurements are noisy, each individual path is subject to a certain amount of random wandering which obscures any structure in its location. By superimposing a large number, the path density gives us some idea of those locations which attract the "attention" of the algorithm. In the case of the ellipse, it can be seen that the path density is greatest at the pole opposite the initial data collection (ϑ = –90°, φ = 0°). This matches the strategy most people take when asked to resolve the ambiguity of a single view, that is to look at the other side.

What is interesting, however, is the anisotropy of the path locations. There is definitely a greater density along the meridians φ = 0° and φ = 180° and a close examination of the corresponding 3D rendition of the paths shows that the scanner spends more time exploring the narrower, more highly curved surfaces of the ellipsoid. The attraction of the scanner to places of high curvature is demonstrated graphically by the exploration paths of the block where it is obvious that scanner spends most of its time collecting data from the edges and the corners. It would appear that information is in some way associated with the curvature of the surface.

In other simulation experiments, we have confirmed that the use of model uncertainty results in faster recovery of accurate model parameters when compared with geometric methods, and that it does so even when model pose and size are arbitrarily changed. An important point here is knowing when exploration is complete, and for this we have developed a notion of coverage $\lceil_s \sigma_D^2(\mathbf{s})/\sigma^2$. Basically, when the model can interpolate the surface everywhere to the same accuracy as the measurements, then it indicates that exploration has recovered all of the structural information. Further exploration does improve model accuracy, but only at a slow rate which is equivalent to the effect of repeating measurements. Details can be found in [20].

4.5 Real-World Complications

Unlike the simulations in Fig. 5, there are further complications in the real world. In the simulations, the model's surface always enclosed the center of the view sphere. When that is not true, the situation is as shown in Fig. 3. The scanner will be unable to leave the upper view sphere region and can only collect data from the top portion of the model's surface. The convergence of det ($\mathbf{C}_n$) will be much slower (proportional to n^{-p}) than if the sensor were free to make measurements of the lower portion of the surface.

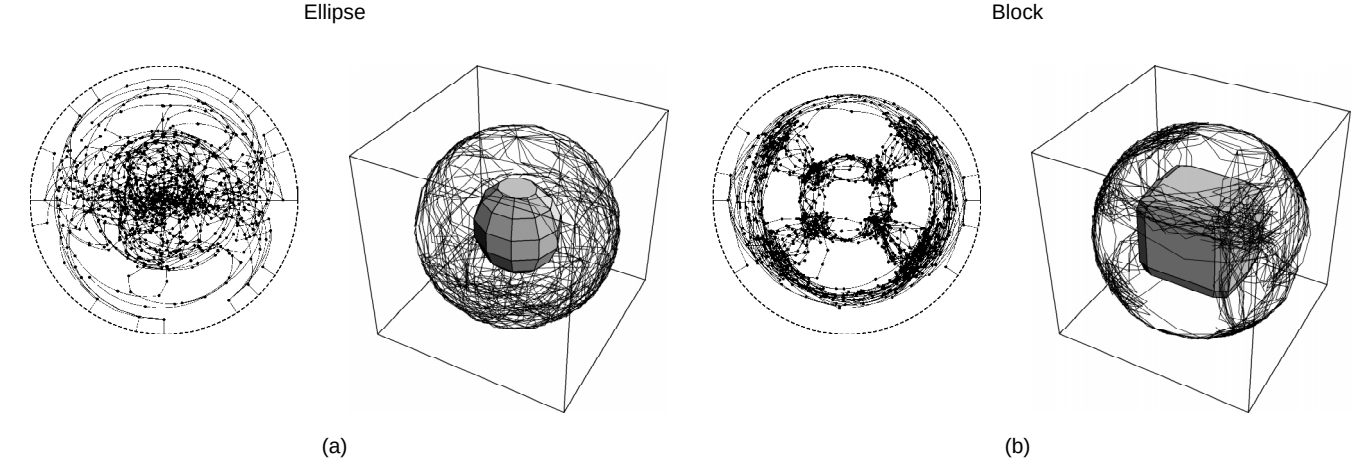


Fig. 5. Forty explorations of an ellipse and a block. The view sphere trajectories of all the explorations are displayed in two forms. (a) View sphere coordinates are shown in polar form, as though the view sphere were seen from the underneath. The view sphere latitude is plotted radially with $\vartheta = -90°$ in the center and $\vartheta = 90°$ at the outer dashed circle. Longitude is the angular coordinate with $\varphi = 0$ being horizontal and to the right. The initial data were scanned from $\vartheta = 90°$ so the small radial lines show the initial 20° step. The dots mark the places from which additional data were scanned and the lines connecting them the great circle paths taken by the scanner. (b) The 3D positions of the paths in (a) are shown on the surface of a transparent view sphere. The model being explored is rendered in the center.

Currently, we solve this problem by always repositioning the view sphere so that its center is at the center of the current model estimate. However, we are also investigating another approach based on the observation that the mapping $\mathbf{x} \mapsto \mathbf{s}$ is a projection of the $\sigma_D^2(\mathbf{s})$ field from the surface of the model to the surface of the view sphere. In the gradient strategy, this projection happens to be discontinuous but there are many others which are not, for example if we project the field radially from the model's center. We have obtained some encouraging preliminary results with model centered projections that adapt to increase the camera step size as σ_D^2/σ^2 decreases.

Problems also arise because the next scanner position is based upon an estimate of where the model's surface should be. In the early iterations, there is usually a large amount of error and it is possible to compute a new location from which no measurement of the true surface is possible. By keeping the step size small, we reduce the chance of this happening. A better approach would be to adjust the scanner travel so it never moves into regions where σ_D^2 is very large. This is exactly the behavior that some of the adaptive projections mentioned in the previous paragraph exhibit.

A difficult problem is that of accessibility. It is rarely the case that there is complete freedom to sample any surface in the scene. Real objects are supported by, embedded within, or occluded by other objects. Real scanners cannot move everywhere. Their physical size prevents them passing along narrow passages, and the device that moves them, e.g., a robot arm, has its own limitations. It will simply not be possible to sample the most uncertain surface in some, and probably most, cases. The successful resolution of these problems would require task specific knowledge, for example, details of the dimensions of the scanner, the robot, and the workspace configuration. A basic tenant of our design is that it be modular and general, so we should not have to customize the gaze planning strategy for each application. We consider the resolution of accessibility problems to be beyond the domain of the gaze planning strategy.

What is important is that the strategy should operate in a way which makes it possible to handle external problems like lack of accessibility. In this regard, our requirement that the sensor only move incrementally is a prudent course of action. If a problem occurs, the sensor will still be close to its last position, and it will be easy to backtrack and recover. There are also practical reasons to prefer small, incremental movements. When the scanner is mounted on a robot arm, it is much simpler to compute a path (and to implement collision avoidance) over a short distance than it is to traverse from one side of the work space to the other. Another important requirement is the need to convert scanner centered 3D coordinates into the scene coordinates used to fit models [22]. Because the position of the sensor is uncertain, we can only do this by the registration of overlapping data sets, and smaller movements let us to keep the area scanned small whilst still maintaining a useful overlap.

5 Exploring

Although we can alleviate many of the above problems, there is always a chance that they will occur, and that task specific knowledge will be required to take corrective action. For this, we need a higher authority. The analogy we use is that the gaze planning strategy plays the role of a *navigator*, and has the specific task of determining what the best heading is at each iteration. The navigator operates under the command of a more general module which we call the *explorer*, and it is this which has the task specific knowledge needed to the verify the operational feasibility of the heading, and to detect and correct the kinds of problems which arise when the heading is followed.

At the heart of the explorer is a servo loop built around the gaze planning strategy. However, the explorer is more than a simple servo loop. It is an executive that delegates

jobs, monitors progress, and makes decisions. The decision as to when to stop is largely determined by the application task requesting the models. There are complications because the specification is application dependent, for example, an object recognition system will want a better knowledge of specific model parameters in order that it may disambiguate two models stored in its data base, and a vehicle might be more interested in the surface location along its proposed path. In principal, however, these criteria are equivalent but different mappings of the parameter covariances.

Handling accessibility constraints should also be dealt with by the explorer. The incremental approach to gaze control helps somewhat but inevitably an exploration path will be forced to terminate because of occlusion or because of the inability of the robot moving the scanner to access the correct location. In these cases, the explorer must take control, override the servo loop, and reinitialize the scanner at a new location.

The gaze planning strategy is also very single minded—it only works upon a single model. When there are multiple models in the scene, it is the explorer which must decide the focus of attention. The strategy employed will depend to a large extent on which models the application task is currently interested in.

A very important task for the explorer is to monitor the behavior of the lower levels of processing. None of these can be guaranteed to operate flawlessly, for reasons ranging from excessively noisy data, to situations in the scene that break the assumptions upon which the algorithms are based. A general "catch-all" way of determining if something has gone wrong is to measure the amount of misfit between the inferred models and the data, i.e., to examine the residual errors. When there is misfit, the residual errors will significantly exceed the expected sensor noise, the servo loop can be aborted, diagnostics run to isolate the source of the error, and remedial action taken to fix the problem where possible. The problem with this is that we must have good models of sensor noise that we may compare statistically with the residual errors. We have investigated this and have found that the classic statistical methods for detecting misfit can be quite sensitive to departures from the theoretical model. To this end, we have devised a sequential estimator of sensor noise which operates within the servo loop. We have shown that this estimator reliably detects misfit even when the sensor noise is not that of the theory, and when the noise level varies during the course of exploration [23].

6 Implementation of an Autonomous Explorer

The concepts of the previous sections have been used to implement an autonomous explorer capable of building an articulated volumetric description of its environment through a sequence of exploratory probes. Fig. 6 shows a block diagram of the resulting implementation. The left side corresponds to a classical model of bottom-up vision in which sensor data are transformed into various levels of representation through successive stages of processing. Data are acquired through a laser range-finding system mounted on the end-effector of an inverted PUMA 560 robot. Curvature based methods are used to find part boundaries and to register data from different views. At the highest level of abstraction, surface regions defined by the part boundaries are described by parametric models, in this case superquadrics [24], [22], [2], [25].

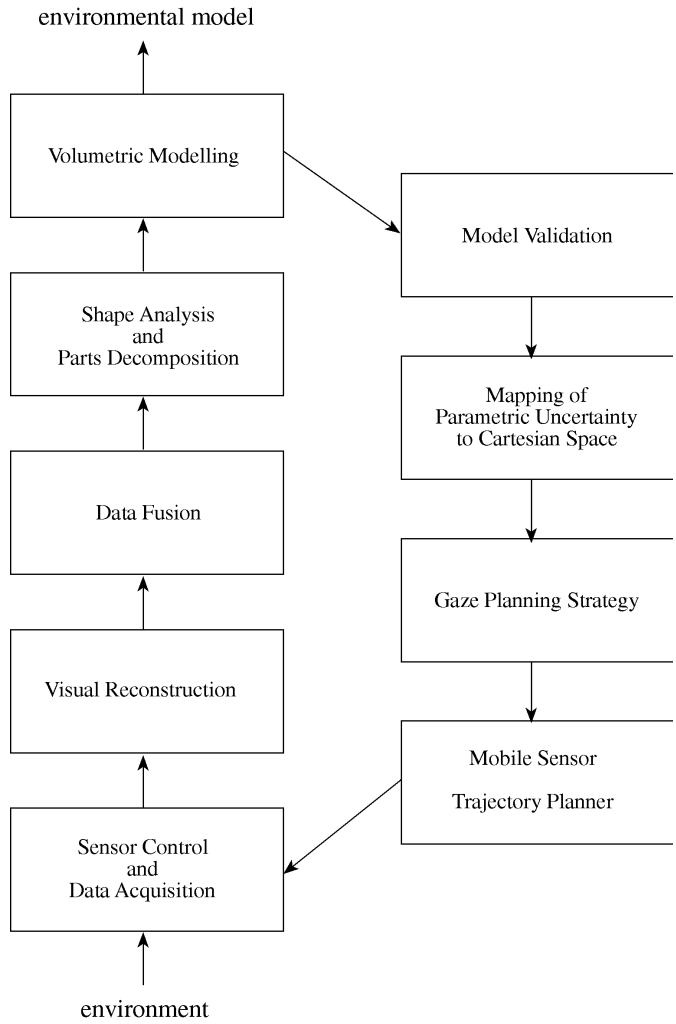
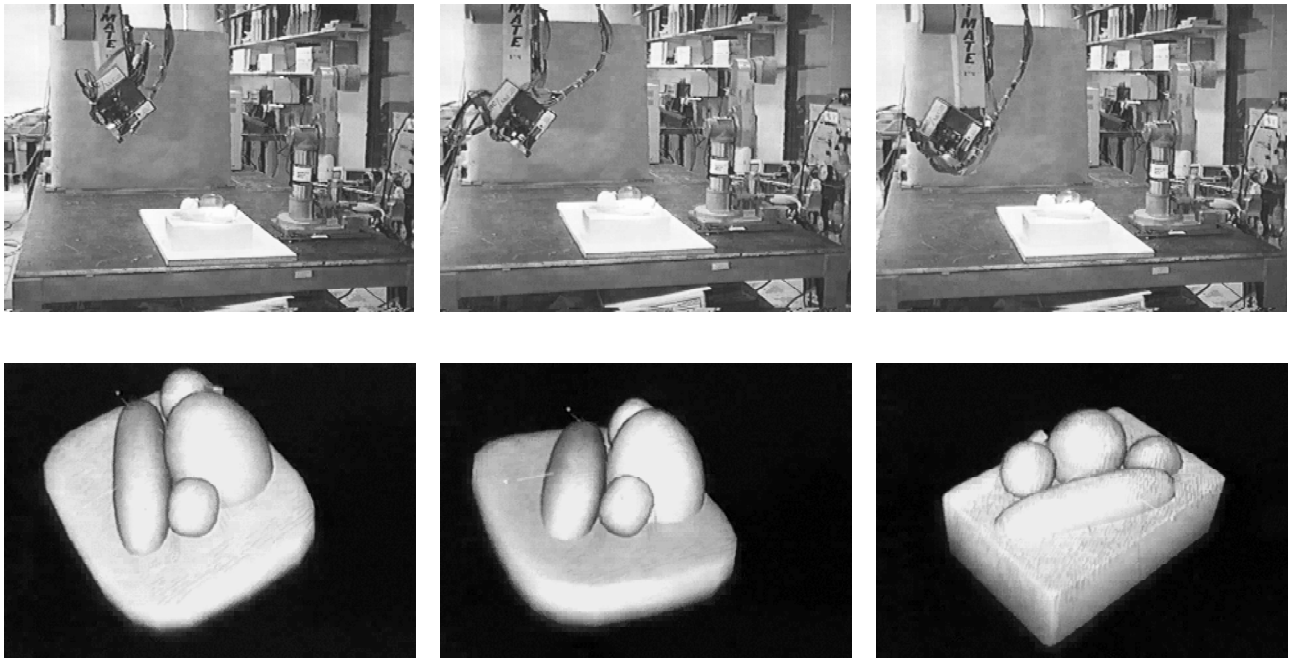


Fig. 6. Process flow in the autonomous explorer. The left hand side of the figure corresponds to a classical bottom-up vision strategy. The right hand side corresponds to feedback derived from parametric uncertainty which is used to close the loop around bottom-up perception.

6.1 Closing the Loop

Much as in the same way that feedback serves to reduce plant uncertainty in a conventional control system, the autonomous explorer uses feedback to minimize the uncertainty of the parametric models used to describe the scene. The right hand side of Fig. 6 shows how this feedback is implemented. A fundamental assumption implicit in the strategy is the validity of the models used as reference, i.e., that a particular model is competent to describe its data in the first place. The sequential estimator of sensor noise discussed earlier in Section 5 is used for this purpose. At the present time, no attempt is made to backtrack in the event that a model is deemed invalid; the exploration process for the model in question is simply reinitialized. The



(a) View 1 (b) View 2 (c) View 3

Fig. 7. Autonomous exploration sequence. (a) The initial viewpoint (top) and corresponding scene model (bottom). (b) Second viewpoint determined from the uncertainties in the initial model. Scene model computed from fusion of data in the first and second views. (c) Final viewpoint and composite scene model computed from all three views.

default strategy is simply to throw away the data corresponding to the model in question and begin again with the data in the current viewpoint. This strategy, while not optimal, works quite well in practice as failures are often a result of segmentation errors.

Gaze planning is based on the view sphere gradient strategy developed for a single model in Section 4.4. However, the problem becomes much more complicated when an object comprised of many parts (models) must be explored by a single mobile sensor. Our initial solution, the one reported in this paper, was to apply a focus of attention mechanism. At each gaze planning iteration, all models are examined and valid ones ranked according to size, magnitude of the uncertainty gradient and distance to the sensor. The "winner" gains control of the sensor for that iteration and a next view is chosen that corresponds to the direction on the view sphere with the largest uncertainty gradient relative to the current position. The exploration process is allowed to run until all models fall below a prescribed error of fit threshold or until a maximum number of exploratory probes have been completed.

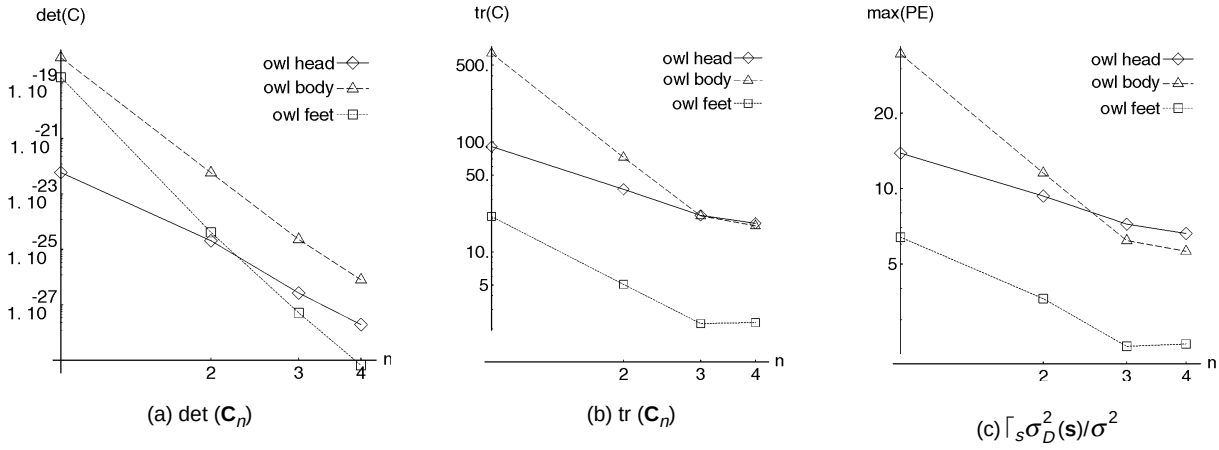
6.2 Experimental Results

The prototype implementation of our exploration system consists of a collection of processes distributed over a network of special and general purpose computers corresponding to the flow diagram shown in Fig. 6. A 128×128 laser range-finder image was collected at each iteration. The total processing time per iteration is data-dependent, but for the example shown here was approximately 30 seconds. Of this, the time taken to plan the trajectory was under 1 second, so it essentially comes for free in these experiments. The majority of the time was taken by the correspondence process used to register data from different views. A more accurate positioning system would speed this step considerably. For all processes except volumetric modeling, computations were distributed between a Silicon Graphics IRIS 4D/35 and an Indigo VX workstation.

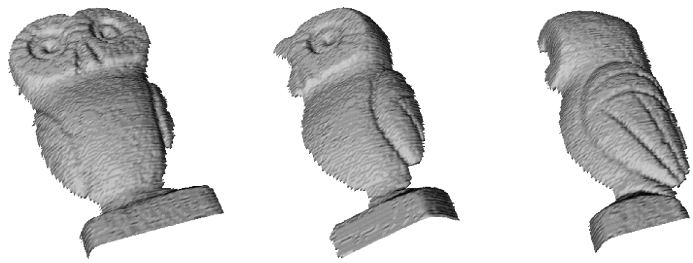
Some qualitative results are shown in Fig. 7. The system starts off by coarsely sampling the workspace until the block and fruit are at the center of its field of view. From this position, models are fitted to data scanned at full resolution (Fig. 7a). All five parts comprising the "object" are correctly localized, but there are significant errors in the positions and shapes of each part. This is about the best that can be expected given only a single view of the scene without additional constraint information.

Next, a second viewpoint is computed based on the uncertainty surface of the part closest to the rangefinder. This new viewpoint is shown in Fig. 7b along with the model computed from the previous and current views. The additional data serves to further constrain the shapes of the fruit, but provides little additional constraint on the wooden block. Using the same part as in the previous iteration, an uncertainty surface is computed and a third viewpoint determined (Fig. 7c). This viewpoint brings the scanner low on the horizon from which more of the wooden block is visible. The resulting model, shown to the right, incorporates data from all three viewpoints and now correctly represents the shape of each part. We have verified that the overall process is stable, producing near identical results when the initial positions of objects in the scene are perturbed.

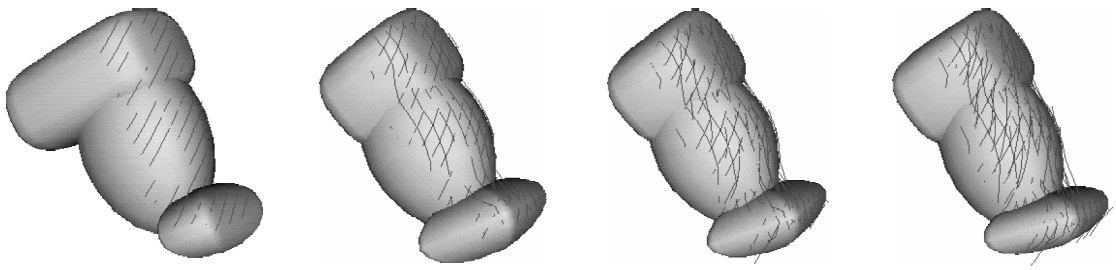
Some quantitative results from exactly the same system are given in Fig. 8. In this case, the object positioned in the



(a) det ($\mathbf{C}_n$) (b) tr ($\mathbf{C}_n$) (c) $\lceil_s \sigma_D^2(\mathbf{s})/\sigma^2$



(d) Range data showing three different views of the owl's surface. The range data are denser than those used in the experiments, and have been artificially shaded.



(e) The recovered superellipsoid part models at each iteration. The range finder scan lines are shown superimposed on the models. The cross hatched pattern indicates data taken from different views.

Fig. 8. Autonomous exploration of owl statuette showing the decrease in (a) det ($\mathbf{C}_n$), (b) tr ($\mathbf{C}_n$), and (c) coverage ($\lceil_s \sigma_D^2(\mathbf{s})/\sigma^2$) [20] over four iterations, some range image views of the owl's surface (d), and the evolution of the part models (e).

work space was a small stone owl. Fig. 8d shows the kind of range data obtained from the scanner, though for this experiment, the actual data used was sampled from grid four times courser, so we should not expect parts to evolve which depend on resolution of finer details (e.g., the wings). A wide field of view was required so that range images from different views could be registered into the same coordinate frame.

Processing of the initial scan resulted in a segmentation of the range data into three separate patches. The superellipsoid part models fitted to each patch are shown in Fig. 8e.1. Some of the range data scan lines can be seen on the surface, and give an idea of the amount of surface covered. Because it was closer than the other two parts, the explorer focused its initial attention on the owl's body to select a new viewpoint. After moving the scanner to the new view, another scan was taken, segmented, then registered with the first. The second set of scan lines crossing the original in Fig. 8e.2 indicate the region from which new data was obtained, and also the success of registration algorithm. The part models, particularly the head, are now more constrained because the second view added additional data on the side of owl. The process continues for another two iterations, with the part models, especially the feet, becoming smaller at each step. Further improvement would require access to the back surfaces, but these are inaccessible because the owl is resting on a table.

All the uncertainty criteria for each part model decrease

at approximately the same rate (Figs. 8a, 8b, 8c) probably because of the wide field of view, and because each part is roughly in the same pose. Under these conditions, similar sensor trajectories would have been obtained with attention focused on any of the models. The slightly faster rate of decrease for the feet can be attributed to the model's smaller size and the larger percentage of its surface covered by the data collected at each iteration. The leveling off of $\int_s \sigma_D^2(\mathbf{s})/\sigma^2$ (and tr (**C**)) after the third iteration is an indication of the inaccessibility of parts of the surface, and of the inability of the collected data to improve knowledge of the surface in those positions.

It is important to note that in both of these experiments many of the assumptions upon which the exploration algorithm is based are not valid. For example, the shape of the owl cannot be described precisely with superellipsoid models—the body is asymmetric, and the head is not even convex. Despite this, exploration performs well and recovers useful models which capture the general size, pose, and shape of each part. What we observe in practice is that the distribution of prediction variance across the surface of the model still makes sense even when the model fits the data badly. It is lower where there is data, and increases rapidly where there is none. A strategy based on this distribution might be suboptimal, but the sensor will still be driven to locations where no data was previously obtained. We have assumed that the a posteriori model parameter distribution is normal, but even gross departures from normality result in reasonable behavior, and in a system which exhibits a high degree of robustness.

7 Conclusions

The results presented in the previous section demonstrate that feedback based on model uncertainty can effectively be used to plan gaze and reliably infer scene descriptions. In particular, we demonstrated how a description comprised of articulated volumetric models could be automatically computed from a sequence of exploratory probes obtained by a mobile laser range-finding system. The resulting models are sufficiently robust to serve as object descriptors for purposes of manipulation and recognition. Because the process is entirely data-driven, the system is well-suited as a basis for artificial perception in unstructured environments. It is also completely-autonomous. Sensor measurement, data fusion, model inference, and gaze planning proceed iteratively either until a stable description of the scene is obtained or a prerequisite amount of data has been collected.

We are currently extending this research in a number of directions. More general purpose models are being investigated which can take into account the dynamic behavior of objects. Ways of implementing backtracking when a model fails to account for newly acquired data are being incorporated into the system. By proceeding in this manner, we hope to eventually learn enough about the general problem to build systems that are truly capable of autonomous exploration.

Acknowledgments

This research was supported in part by the Natural Sciences and Engineering Research Council of Canada under grant OGPIN 011 and the Institute for Robotics and Intelligent Systems, a Network of Centres of Excellence. The authors also wish to thank G. Soucy for computational support and S.W. Zucker for helpful discussions and comments throughout the course of this work.

References

[1] T.H. Choi, H. Delingette, M. DeLusie, Y. Hsin, M. Hebert, and K. Ikeuchi, "A Perception and Manipulation System for Collecting Rock Samples," *Proc. Fourth Ann. Space Operations, Applications, and Research Symp.: SOAR 90*, Albuquerque, N.M., June 1990.

[2] F.P. Ferrie, J. Lagarde, and P. Whaite, "Darboux Frames, Snakes, and Super-Quadrics: Geometry from the Bottom Up," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 8, pp. 771-784, Aug. 1993.

[3] A. Pentland, "Recognition by Parts," *ICCV87* [26], pp. 612-620.

[4] W. Cheung, F.P. Ferrie, G. Carayannis, and J.B. Edwards, "Rockpile Surface Decomposition: Machine Vision in Mining," *Proc. Canadian Conf. Industrial Automation*, Montréal, Québec, June 1-3, 1992.

[5] P. Whaite and F.P. Ferrie, "From Uncertainty to Visual Exploration," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 10, pp. 1,038-1,049, Oct. 1991.

[6] P. Whaite and F.P. Ferrie, "Uncertain Views," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, Champaign, Ill., pp. 3-9, June 15-18, 1992.

[7] R. Bajcsy, "Active Perception," *Proc. IEEE*, vol. 76, no. 8, pp. 996-1,005, 1988.

[8] R. Bajcsy and M. Campos, "Active and Exploratory Perception," *Computer Vision, Graphics, and Image Processing*, vol. 56, no. 1, pp. 31-40, July 1992.

[9] C.I. Connolly, "The Determination of Next Best Views," *Proc. IEEE Int'l Conf. Robotics Automation*, pp. 432-435, 1985.

[10] N. Ahuja and J. Veenstra, "Generating Octrees from Object Silhouettes in Orthographic Views," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, no. 2, pp. 137-149, Feb. 1989.

[11] K. Tarabanis and R.Y. Tsai, "Computing Viewpoints that Satisfy Optical Constraints," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, Lahaina, Maui, Hawaii, pp. 152-158, June 1993.

[12] J. Maver and R. Bajcsy, "Occlusions as a Guide for Planning the Next View," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 5, pp. 417-433, May 1993.

[13] J. Aloimonos and A. Bandyopadhyay, "Active Vision," *ICCV87 Proc. First Int'l Conf. Computer Vision*, London, U.K., IEEE CS Press, pp. 35-54, June 1987.

[14] K.N. Kutulakos and C.R. Dyer, "Global Surface Reconstruction by Purposive Control of Observer Motion," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, Seattle, Wash., pp. 331-338, June 1994.

[15] L. Birnbaum, M. Brand, and P. Cooper, "Looking for Trouble: Using Casual Semantics to Direct Focus of Attention," *ICCV93 Proc. Fourth Int'l Conf. Computer Vision*, Berlin, Germany, IEEE CS Press, pp. 49-56, May 11-14, 1993.

[16] G.D. Hager, *Task-Directed Sensor Fusion and Planning: A Computational Approach*. Kluwer Academic Publishers, 1990.

[17] A.M. Mood and F.A. Graybill, *Introduction to the Theory of Statistics*. New York: McGraw-Hill Book Co., Inc., 1963.

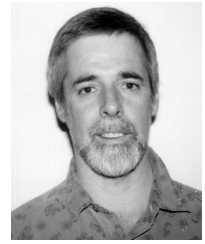
[18] D.J.C. MacKay, "Bayesian Methods for Adaptive Models," PhD thesis, California Inst. of Technology, Pasadena, 1992.

[19] V.V. Fedorov, *Theory of Optimal Experiments*. New York: Academic Press, 1972.

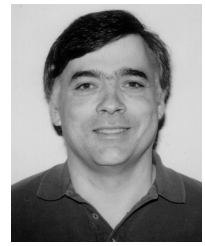
[20] P. Whaite and F.P. Ferrie, "Autonomous Exploration: Driven by Uncertainty," Technical Report TR-CIM-93-17, Center for Intelligent Machines, McGill Univ., Montréal, Québec, Canada, 1993, available via ftp at ftp.cim.mcgill.ca in /pub/3d/papers/tr-cim-93-17.ps.gz.

[21] H.W. Sorenson, "Least-Squares Estimation: From Gauss to Kalman," *Kalman Filtering: Theory and Application*, H.W. Sorenson, ed., IEEE CS Press, 1985.

[22] G. Soucy, "View Correspondence Using Curvature and Motion Consistency," master's thesis, Dept. of Electrical Eng., McGill Univ., 1992.
[23] P. Whaite and F.P. Ferrie, "Active Exploration: Knowing When We're Wrong," *ICCV93 Proc. Fourth Int'l Conf. Computer Vision*, Berlin, Germany, IEEE CS Press, pp. 41-48, May 11-14, 1993; "On the Sequential Determination of Model Misfit," Technical Report TR-CIM-94-11, Center for Intelligent Machines, McGill Univ., Montréal, Québec, Canada, 1993, available via ftp at ftp.cim.mcgill.ca in /pub/3d/papers/tr-cim-93-17.ps.gz.
[24] F.P. Ferrie, S. Mathur, and G. Soucy, "Feature Extraction for 3D Model Building and Object Recognition," *3D Object Recognition Systems*, A. Jain and P. Flynn, eds., Elsevier, Amsterdam, 1993.
[25] A. Lejeune and F.P. Ferrie, "Partitioning Range Images Using Curvature and Scale," *Computer Vision, Graphics, and Image Processing*, vol. 64, no. 2, pp. 230-247, Sept. 1996.
[26] *Proc. First Int'l Conf. Computer Vision*, London, U.K., June 1987, IEEE CS Press.
[27] *Proc. Fourth Int'l Conf. Computer Vision*, Berlin, Germany, May 11-14, 1993, IEEE CS Press.
[28] P. Whaite and F.P. Ferrie, "On the Sequential Determination of Model Misfit," Technical Report TR-CIM-94-11, Center for Intelligent Machines, McGill Univ., Montréal, Québec, Canada, 1993, available via ftp at ftp.cim.mcgill.ca in /pub/3d/papers/tr-cim-93-17.ps.gz.



Peter Whaite received the BEng (Elect) degree with honors from the University of New South Wales, Sydney, Australia, in 1973. From 1973 until 1977, he was with the Department of Biological Technology at the University of New South Wales working on the optimal control of chemostats. In 1982, he received the MSc degree from the Department of Geophysics and Astronomy at the University of British Columbia, Vancouver, Canada, for the design of an automated system to prepare and analyze oxygen isotope samples. He remained there until 1985 where he worked on several geophysical instrumentation projects. In 1988, he began his PhD on autonomous exploration with the Artificial Perception Laboratory, Centre for Intelligent Machines, McGill University, Montréal, Canada. He now works there as a research scientist engaged in applications of his thesis topic.



Frank P. Ferrie received the BEng, MEng, and PhD degrees in electrical engineering from McGill University in Montréal in 1978, 1980, and 1986, respectively. During 1985 and 1986, he served as manager of engineering for Servo-Robot, Inc. in Boucherville, Québec. On returning to McGill in 1987, he was appointed as an attaché de recherche in the McGill Research Centre for Intelligent Machines, and in 1990 as assistant professor in the Department of Electrical Engineering. His research interests are in computer vision, primarily in the areas of two- and three-dimensional shape analysis, active perception, dynamic scene analysis, and machine vision. Dr. Ferrie is currently an associate professor in the Department of Electrical Engineering at McGill University and a principal investigator in the Institute for Robotics and Intelligent Systems, a Canadian Network of Centres of Excellence. He is also a member of the IEEE and the Computer Society.