

HOMING USING COMBINATIONS OF MODEL VIEWS

Ronen Basri

Dept of Applied Math
The Weizmann Institute of Science
Rehovot 76100 Israel

Ehud Rivlin

Computer Vision Laboratory
Center for Automation Research
University of Maryland
College Park, MD 20742-3411

Abstract

A method for *homing* in an indoor environment is presented. Homing is the process by which an autonomous system guides itself to a particular location on the basis of sensory input [8]. The proposed method proceeds in two steps. In the first step the robot attempts to recognize the environment, and in the second step the robot navigates within the recognized environment toward the target location. A system for *Localization* (recognizing the environment) and *positioning* (computing the exact coordinates of a robot in the environment) is used. The system represents scenes as sets of 2D views and predicts the appearance of novel views by linear combinations of the model views. Two methods for directing the robot to the target position are proposed. The first approach is based on recovering the exact coordinates of the robot in the environment. Homing is carried out as a direct result of this recovery. The second approach uses a predetermined homing pattern to execute a simpler algorithm for the process. At each stage the direction to the target is computed directly from the image. Knowledge of the transformation between the model views is not required in this approach.

1 Introduction

Homing is the process by which an autonomous system guides itself to a particular location on the basis of sensory input [8]. Two basic tasks in the process are localization and positioning. *Localization* is the act of recognizing the environment, that is, assigning consistent labels to different locations, and *positioning* is the act of computing the position of the robot in a given environment. Positioning is a task complementary to localization, in the sense that position (e.g., "1.5 meters northwest of table T ") is often specified in a place-specific coordinate system ("in room 911"). In this paper we suggest a method for homing based on these two processes. First localization is attempted, then given the environment homing is carried out based on a positioning process.

Previous studies have examined the problems of lo-

calization and positioning under a variety of conditions, defined by the kind of sensor(s) employed, the nature of the environment, and the representations used. We can distinguish between active and passive sensing, indoor and outdoor navigation tasks, and metric and topological representations. The metric approach attempts to utilize a detailed geometric description of the environment, while the topological approach uses a more qualitative description including a graph with nodes representing places and arcs representing sequences of actions that would result in moving the robot from one node to another.

In the paper we consider a robot that uses a passive sensor, vision, in an indoor environment. The environment cannot be changed by the robot to improve its performance; neither beacons nor floor or wall markings are employed. The paper addresses both the localization and the positioning problems, and then presents a method for homing based on their solution. The method, based on the linear combinations scheme of [14], represents scenes by sets of their 2D images. Localization is achieved by comparing the observed image to linear combinations of model views. Two solutions for the homing process are then presented, based on two different approaches for the positioning process. In the first approach the position of the robot is computed by analyzing the coefficients of the linear combination that aligns the model to the image. The direction to the target position is determined by comparing the computed robot position with the target position. Recovering the exact position of the robot, as well as the target position, is required in this approach. In a second, qualitative approach a predetermined homing pattern is used to compute the direction to the target. Exact coordinates of the robot in the environment are not required in this approach.

The rest of the paper is organized as follows. The next section describes different solutions to the localization and positioning problems. The method of localization using linear combinations of model views is described in Section 3. The two solutions for the homing process are presented in Section 4 and 5. Experimental results are presented in section 6.

2 The Problem

Localization and positioning from visual input are defined in the following way: Given a familiar environment, identify the observed environment, and then find your position in that environment. Localization resembles the task of object recognition, with objects replaced by scenes. Once localization is accomplished, positioning can be performed.

One problem a system for localization and positioning should address is the variability of images due to viewpoint changes. The inexactness of practical systems makes it difficult for a robot to return to a specified position on subsequent visits. The visual data available to the robot between visits varies in accordance with the viewing position of the robot. Consequently, a localization system should be able to recognize scenes from different positions and orientations.

Another problem is that of changes in the scene. At subsequent visits the same place may look different due to changes in the arrangement of the objects, the introduction of new objects, and the removal of others. In general, some objects tend to be more static than others. While chairs and books are often moved, tables, closets, and pictures tend to change their position much less, and walls are almost guaranteed to be static. Static cues naturally are more reliable than mobile ones. Confining the system to static cues, however, may in some cases result in failure to recognize the scene due to insufficient cues. The system should therefore attempt to rely on static cues, but should not ignore the dynamic cues.

Solutions to the problem of localization from visual data require a large memory and heavy computation. Existing systems often try to reduce this cost by using sparse representations and by exploiting contextual information. Sparse representations are introduced in [7, 11]. Mataric [7] represents scenes as sequences of landmarks (such as walls, doors, etc.) extracted by tracing the boundaries of the scene using a sonar and a compass. Metric information of and between the landmarks is not stored. Sarachik [11] recognizes a room by its dimensions, which are measured by identifying and locating the top corners of the room using stereo data (obtained from four cameras). In both cases the representation is very sparse, and the scene is therefore often ambiguous.

Richer representations are used in [1] where higher success rates are reported. Braunegg [1] represents the scene by an occupancy table, a 2D bit array which contains a 1 at every location occupied by some object. The table is constructed by taking stereo pictures covering 360° from the middle of the room and projecting the obtained 3D data onto the floor. The method suffers from loss of information due to the projection onto the floor.

Systems that use the full information provided by the image (e.g., [3, 2]) usually rely on contextual information to avoid scanning all the models in the memory and to reduce the computational cost of comparing a model to the image. The system follows a predetermined path, so that the identity of each visited location is known in advance, and localization becomes a verification prob-

lem. Path continuity in many cases is essential, and the so-called "drop-off" problem is not addressed. The emphasis in these systems is on positioning, which is used to keep the robot on the path. It is typical for these systems (e.g., [2, 3]) to use a full 3D model of the environment.

The method presented in this paper represents the environment using a set of edge maps. Localization and positioning are achieved by comparing images of the environment to linear combinations of the model views. The method uses rich visual information to represent the scene. The system is flexible. In many cases it is capable of recognizing its location from one image only (360° coverage is not required). When one image is not sufficient, additional images can be acquired to solve the localization problem. Context can be used to determine the order of comparison of the models to the observed image and to increase the confidence of a given match, but context is not essential: the system can also, by performing more extensive computations, solve the "drop-off" problem.

3 Localization

In what follows we present a method for localization as a first stage in the homing process. The problem of localization is defined as follows: given P , a 2D image of a place, and $\mathcal{M}$, a set of stored models, find a model $M^i \in \mathcal{M}$ that matches P . A common approach to handling the problem of recognition from different viewpoints is by comparing the stored models to the observed environment after the viewpoint is recovered and compensated for. This approach, called *alignment*, is used in a number of studies of object recognition [4, 5, 6, 12, 13]. We apply the alignment approach to the problem of localization. The system described below uses the "Linear Combinations" (LC) scheme, which was suggested by Ullman and Basri [14].

A LC scheme for the problem of localization is as follows: The environment is modeled by a set of images with correspondence between the images. For example, a spot can be modeled by two of its corresponding views. Given an image, we construct two view vectors from the feature points in the image, one contains the x -coordinates of the points, and the other contains the y -coordinates of the points. The appearance of a novel view of the scene is predicted by applying linear combinations to the stored views. The predicted appearance is then compared with the actual image, and the object is recognized if the two match. The advantage of this method is twofold. First, viewer-centered representations are used rather than object-centered ones, namely, models are composed of 2D views of the observed scene; second, novel appearances are predicted in a simple and accurate way (under weak perspective projection).

Formally, given P , a 2D image of a scene, and $\mathcal{M}$, a set of stored models, the objective is to find a model $M^i \in \mathcal{M}$ such that $P = \sum_{j=1}^k \alpha_j M_j^i$ for some constants $\alpha_j \in \mathcal{R}$. It has been shown that this scheme accurately predicts the appearance of rigid objects under weak perspective projection (orthographic projection and scale).

The limitations of this projection model are discussed in [10].

More concretely, let $p_i = (x_i, y_i, z_i)$, $1 \leq i \leq n$, be a set of n points in the environment. Under weak perspective projection, the position $p'_i = (x'_i, y'_i)$ of these points in the image are given by

$$\begin{aligned} x'_i &= sr_{11}x_i + sr_{12}y_i + sr_{13}z_i + t_x \\ y'_i &= sr_{21}x_i + sr_{22}y_i + sr_{23}z_i + t_y \end{aligned} \quad (1)$$

where r_{ij} are the components of a 3×3 rotation matrix, and s is a scale factor. Rewriting this in vector equation form we obtain

$$\begin{aligned} \mathbf{x}' &= sr_{11}\mathbf{x} + sr_{12}\mathbf{y} + sr_{13}\mathbf{z} + t_x\mathbf{1} \\ \mathbf{y}' &= sr_{21}\mathbf{x} + sr_{22}\mathbf{y} + sr_{23}\mathbf{z} + t_y\mathbf{1} \end{aligned} \quad (2)$$

where $\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{x}', \mathbf{y}' \in R^n$ are the vectors of x_i, y_i, z_i, x'_i and y'_i coordinates respectively, and $\mathbf{1} = (1, 1, \dots, 1)$. Consequently,

$$\mathbf{x}', \mathbf{y}' \in \text{span}\{\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{1}\} \quad (3)$$

or, in other words, $\mathbf{x}'$ and $\mathbf{y}'$ belong to a four-dimensional linear subspace of R^n . A four-dimensional space is spanned by any four linearly independent vectors of the space. Two views of the scene supply four such vectors [9, 14]. Denote by $\mathbf{x}_1, \mathbf{y}_1$ and $\mathbf{x}_2, \mathbf{y}_2$ the location vectors of the n points in the two images; then there exist coefficients a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4 such that

$$\begin{aligned} \mathbf{x}' &= a_1\mathbf{x}_1 + a_2\mathbf{y}_1 + a_3\mathbf{x}_2 + a_4\mathbf{1} \\ \mathbf{y}' &= b_1\mathbf{x}_1 + b_2\mathbf{y}_1 + b_3\mathbf{x}_2 + b_4\mathbf{1} \end{aligned} \quad (4)$$

Localization is achieved by recovering the linear combination that aligns the model to the observed image. The coefficients are determined using four model points and their corresponding image points by solving a linear set of equations. More details about the process can be found in [10]. After localization is achieved positioning can begin.

4 Homing based on exact positioning

Positioning is the problem of recovering the position of the robot. In this section we describe a method for recovering the exact coordinates of the robot in its environment. This position is then used in a straightforward manner to direct the robot to its target position.

The idea is the following. We assume a model composed of two images, P_1 and P_2 ; their relative position is given. Given a novel image P' , we first align the model with the image (i.e., localization). By considering the coefficients of the linear combination the robot's position relative to the model images is recovered. To recover the absolute position of the robot in the room the absolute positions of the model views should also be provided.

Assuming P_2 is obtained from P_1 by a rotation R , translation $t = (t_x, t_y)$, and scaling s , the coordinates of a point in P' , (x', y') , can be written as linear combinations of the corresponding model points in the following way:

$$\begin{aligned} x' &= a_1x_1 + a_2y_1 + a_3x_2 + a_4 \\ y' &= b_1x_1 + b_2y_1 + b_3x_2 + b_4 \end{aligned} \quad (5)$$

Since

$$x_2 = sr_{11}x_1 + sr_{12}y_1 + sr_{13}z_1 + t_x \quad (6)$$

we obtain

$$\begin{aligned} x' &= (a_1 + a_3sr_{11})x_1 + (a_2 + a_3sr_{12})y_1 + (a_3sr_{13})z_1 + (a_3t_x + a_4) \\ y' &= (b_1 + b_3sr_{11})x_1 + (b_2 + b_3sr_{12})y_1 + (b_3sr_{13})z_1 + (b_3t_x + a_4) \end{aligned} \quad (7)$$

Using these equations we can derive all the parameters of the transformation between the model and the image. Assume the image is obtained by a rotation U , translation t_n , and scaling s_n . Using the orthonormality constraint we can first derive the scale factor

$$\begin{aligned} s_n^2 &= (a_1 + a_3sr_{11})^2 + (a_2 + a_3sr_{12})^2 + (a_3sr_{13})^2 \\ &= a_1^2 + a_2^2 + a_3^2s^2 + 2a_3s(a_1r_{11} + a_2r_{12}) \end{aligned} \quad (8)$$

Using Equations (7) and (8), by deriving the components of the translation vector, t_n , we can obtain the position of the robot in the image relative to its position in the model views:

$$\begin{aligned} \Delta x &= a_3t_x + a_4 \\ \Delta y &= b_3t_y + b_4 \\ \Delta z &= f\left(\frac{1}{s_n} - \frac{1}{s}\right) \end{aligned} \quad (9)$$

Note that Δz is derived from the change in scale of the object. The rotation matrix U between P_1 and P' is given by

$$\begin{aligned} u_{11} &= \frac{a_1 + a_3sr_{11}}{s_n} & u_{12} &= \frac{a_2 + a_3sr_{12}}{s_n} \\ u_{13} &= \frac{a_3sr_{13}}{s_n} & u_{21} &= \frac{b_1 + a_3sr_{21}}{s_n} \\ u_{22} &= \frac{b_2 + a_3sr_{22}}{s_n} & u_{23} &= \frac{b_3sr_{23}}{s_n} \end{aligned} \quad (10)$$

As was already mentioned, the position of the robot is computed here relative to the position of the camera when the first model image, P_1 , was acquired. Δx and Δz represent the motion of the robot from P_1 to P' , and the rest of the parameters represent its 3D rotation and elevation. This computation requires the transformation parameters between the model views, P_1 and P_2 .

Based on this derivation a simple method for homing can be implemented. The process requires the coordinates of the current position of the robot, as were computed using Eq. (9) and (10), and the coordinates of the target position. A straightforward computation provides the desired direction to the target. The robot then advances in that direction, and the computation is repeated to correct the heading. This iterative process is essential for avoiding errors due to sensory noise, assumptions about the projection, and mechanical errors.

5 Behavior based homing

In the former section we presented a method for homing using exact positioning. In this section we are interested in a more qualitative approach. Under this approach position is not computed. Instead, the robot observes the environment and extracts only the direction to the target location. Unlike the exact approach, the method presented here does not require the recovery of the transformation between the model views.

We assume we are given with a model of the environment together with a target image (which serves as the *homing pattern*). The robot is allowed to take new images as it is moving towards the target. We assume a horizontally moving platform. (In other words, we assume three degrees of freedom rather than six; the robot is allowed to rotate around the vertical axis and translate horizontally.) Below we give a simple computation that determines a path which terminates in the target location. At each time step the robot acquires a new image and aligns it with the model. By comparing the alignment coefficients with the coefficients for the target image the robot determines its next step. The algorithm is divided into two stages. In the first stage the robot fixates on one identifiable point and moves along a circular path around the fixation point until the line of sight to this point coincides with the line of sight to the corresponding point in the target image. In the second stage the robot advances forward or retreats backward until it reaches the target location.

Given a model composed of two images, P_1 and P_2 , P_2 is obtained from P_1 by a rotation about the Y -axis by an angle α , horizontal translation t_x , and scale factor s . and given a target image P_t , P_t is obtained from P_1 by a similar rotation by an angle θ , translation t_t , and scale s_t . Using Eq. (4) the position of a target point (x_t, y_t) can be expressed as

$$\begin{aligned} x_t &= a_1 x_1 + a_3 x_2 + a_4 \\ y_t &= b_2 y_1 \end{aligned} \quad (11)$$

(The rest of the coefficients are zero since the platform moves horizontally.) In fact, the coefficients are given by

$$\begin{aligned} a_1 &= \frac{s_t \sin(\alpha - \theta)}{\sin \alpha} \\ a_3 &= \frac{s_t \sin \theta}{s \sin \alpha} \\ a_4 &= t_t - \frac{t_x s_t \sin \theta}{s \sin \alpha} \\ b_2 &= s_t \end{aligned} \quad (12)$$

(The derivation is given in [10])

At every time step the robot acquires an image and aligns it with the above model. Assume the image P_p is obtained as a result of a rotation by an angle ϕ , translation t_p , and scale s_p . The position of a point (x_p, y_p) is expressed by

$$\begin{aligned} x_p &= c_1 x_1 + c_3 x_2 + c_4 \\ y_p &= d_2 y_1 \end{aligned} \quad (13)$$

where the coefficients are given by

$$\begin{aligned} c_1 &= \frac{s_p \sin(\alpha - \phi)}{\sin \alpha} \\ c_3 &= \frac{s_p \sin \phi}{s \sin \alpha} \\ c_4 &= t_p - \frac{t_x s_p \sin \phi}{s \sin \alpha} \\ d_2 &= s_p \end{aligned} \quad (14)$$

The step performed by the robot is determined by

$$\delta = \frac{c_1}{c_3} - \frac{a_1}{a_3} \quad (15)$$

That is,

$$\begin{aligned} \delta &= \frac{s \sin(\alpha - \phi)}{\sin \phi} - \frac{s \sin(\alpha - \theta)}{\sin \theta} \\ &= s \sin \alpha (\cot \phi - \cot \theta) \end{aligned} \quad (16)$$

The robot should now move so as to reduce the absolute value of δ . The direction of motion depends on the sign of α . The robot can deduce the direction by moving slightly to the side and checking if this motion results in an increase or decrease of $|\delta|$. The motion is defined as follows. The robot moves to the right (or to the left, depending on which direction reduces $|\delta|$) by a step Δx .

A new image P_n is now acquired, and the fixated point is located in this image. Denote its new position by x_n . Since the motion is parallel to the image plane the depth values of the point in the two views, P_p and P_n , are identical. We now want to rotate the camera so as to return the fixated point to its original position. The angle of rotation, β , can be deduced from the equation

$$x_p = x_n \cos \beta + \sin \beta \quad (17)$$

This equation has two solutions. We chose the one that counters the translation (namely, if translation is to the right, the camera should rotate to the left), and that keeps the angle of rotation small. In the next time step the new picture P_n replaces P_p and the procedure is repeated until δ vanishes. The resulting path is circular around the point of focus.

Once the robot arrives at a position for which $\delta = 0$ (namely, its line of sight coincides with that of the target image, and $\phi = \theta$) it should now advance forward or retreat backward to adjust its position along the line of sight. Several measures can be used to determine the direction of motion; one example is the term c_1/a_1 which satisfies

$$\frac{c_1}{a_1} = \frac{s_p}{s_t} \quad (18)$$

when the two lines of sight coincide. The objective at this stage is to bring this measure to 1.

6 Experiments

We have implemented the localization part and applied it to images taken in an indoor environment. Images of two offices, A and B, that have similar structures were taken using a Panasonic camera with a focal length of 700 pixels. Semi-static objects, such as heavy furniture and pictures, were used to distinguish between the offices. Figure 1 shows two model views of office A. The views were taken at a distance of about 4m from the wall. Correspondences were picked manually. The results of aligning the model views to images of the two offices are presented in Figure 2. The left image contains an overlay of a predicted image (the thick white lines), constructed by linearly combining the two views, and an actual image of office A. A good match between the two was achieved. The right image contains an overlay of a predicted image constructed from a model of office B and an image of office A. Because the offices share a similar structure the static cues (the wall corners) were perfectly aligned. The semi-static cues, however, did not match any features in the image.



Figure 1: Two model views of office A.

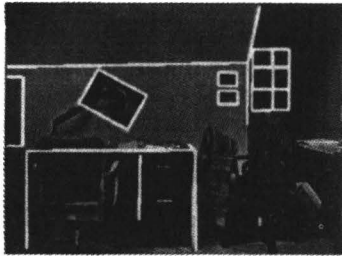


Figure 2: Matching a model of office A to an image of office A (left), and matching a model of office B to the same image (right)

Another set of images was taken in a corridor. Here, because of the deep structure of the corridor, perspective distortions are noticeable. Nevertheless, the alignment results still demonstrate an accurate match in large portions of the image. Figure 3 shows two model views of the corridor. Figure 4 (left) shows an overlay of a linear combination of the model views with an image of the corridor. It can be seen that the parts that are relatively distant align perfectly. Figure 4 (right) shows the matching of the corridor model with an image obtained by a relatively large motion (about half of the corridor length). Because of perspective distortions the relatively near features no longer align (e.g., the near door edges). The relatively far edges, however, still match.

The experimental results demonstrate that the method achieves accurate localization in many cases.

7 Conclusions

An approach for homing based on localization and positioning was introduced. The approach is based on representing the scene as a set of 2D views and predicting the appearance of novel views by linear combinations of the model views. The presented method proceeds in two steps. In the first step the robot attempts to recognize the environment, and in the second step the robot navigates within the recognized environment toward the target location. Two methods for directing the robot to the target position were proposed. The first approach is based on recovering the exact coordinates of the robot in its environment. Homing is carried out as a direct result of this recovery. The second approach uses a predetermined homing pattern to execute a simpler algorithm for the process. At each stage the direction to the target is computed directly from the image. Knowledge of the transformation between the model views is not required in this approach.

The method presented in this paper has several ad-

vantages over existing methods. It uses relatively rich representations; the representations are 2D rather than 3D, and localization can be done from a single 2D view only. The same basic method is used in both the localization and positioning problems. Future work includes experiment the full homing process, handling the problem of acquisition and maintenance of models, developing efficient and robust algorithms for solving the aspects of the correspondence problem needed for this task, and building maps using visual input.

References

- [1] D. J. Braunegg. Marvel—A system for recognizing world locations with stereo vision. *AI-TR-1229*, MIT, 1990.
- [2] N. Ayache and O. D. Faugeras. Maintaining representations of the environment of a mobile robot, *IEEE Trans. on Robotics and Automation*, Vol. 5, pp. 804–819, 1989.
- [3] C. Fennema, A. Hanson, E. Riseman, R. J. Beveridge, and R. Kumar. Model-directed mobile robot navigation. *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 20, pp. 1352–1369, 1990.
- [4] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM*, Vol. 24, pp. 381–395, 1981.
- [5] D. P. Huttenlocher and S. Ullman. Object recognition using alignment. *Proc. 1st Int. Conf. on Computer Vision*, London, UK, pp. 102–111, 1987.
- [6] D. G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Robotics Research Technical Report 202*, Courant Institute of Math. Sciences, New York University, 1985.

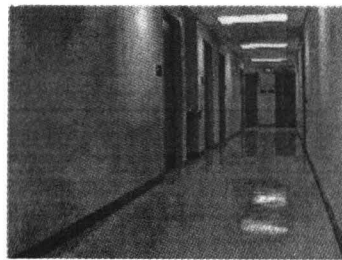


Figure 3: Two model views of a corridor.

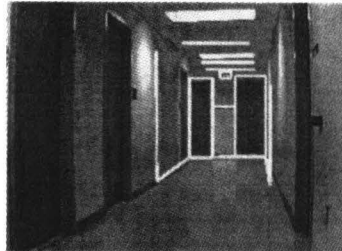
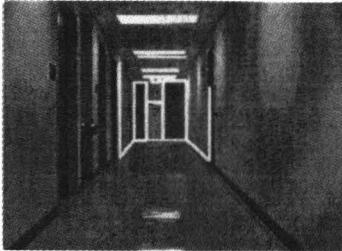


Figure 4: Matching the corridor model with two images of the corridor. The right image was obtained by a relatively large motion forward (about half of the corridor length) and to the right.

- [7] M. J. Mataric. Environment learning using a distributed representation. *Proc. Int. Conf. on Robotics and Automation*, Cincinnati, OH, 1990.
- [8] R. N. Nelson. Visual homing using an associative memory. *DARPA Image Understanding Workshop*, pp. 245-262, 1989.
- [9] T. Poggio. 3D object recognition: on a result by Basri and Ullman. *Technical Report 9005-03, IRST*, Povo, Italy, 1990.
- [10] E. Rivlin and R. Basri, Localization and Positioning using Combinations of Model Views. A.I. Memo 1376, M.I.T., 1992.
- [11] K. B. Sarachik. Visual navigation: constructing and utilizing simple maps of an indoor environment. *AI-TR-1113, MIT*, 1989.
- [12] D. W. Thompson and J. L. Mundy. Three dimensional model matching from an unconstrained viewpoint. *Proc. Int. Conf. on Robotics and Automation*, Raleigh, NC, pp. 208-220, 1987.
- [13] S. Ullman. Aligning pictorial descriptions: an approach to object recognition. *Cognition*, Vol. 32, pp. 193-254, 1989.
- [14] S. Ullman and R. Basri. Recognition by linear combinations of models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 13, pp. 992-1006, 1991.