MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1376 September, 1992

Localization and Positioning using Combinations of Model Views

Ehud Rivlin and Ronen Basri

Abstract

A method for localization and positioning in an indoor environment is presented. *Localizat*
is the act of recognizing the environment, and *positioning* is the act of computing the e
coordinates of a robot in the environment. The method is based on representing the sc
as a set of 2D views and predicting the appearance of novel views by linear combinatio
the model views. The method accurately approximates the appearance of scenes under weak
perspective projection. Analysis of this projection as well as experimental results dem
that in many cases this approximation is sufficient to accurately describe the scene.
orthographic approximation is invalid, either a larger number of models can be acquired
iterative solution to account for the perspective distortions can be employed.

The presented method has several advantages over existing methods. It uses relatively
representations, the representations are 2D rather than 3D, and localization can be done
a single 2D view only. The same principal method is applied both for the localization as
as the positioning problems, and a simple algorithm for *repositioning*, the task of returni
a previously visited position defined by a single view, is derived from this method.

This report describes research done at the Massachusetts Institute of Technology withi
Artificial Intelligence Laboratory and the McDonnell-Pew Center for Cognitive Neuroscie
Support for the laboratory's artificial intelligence research is provided in part by the A
Research Projects Agency of the Department of Defense under Office of Naval Research con-
tract N00014-91-J-4038. Ronen Basri is supported by the McDonnell-Pew and the Rothchil
postdoctoral fellowships. Ehud Rivlin is at the University of Maryland, College Park, M

1 Introduction

Basic tasks in autonomous robot navigation are localization and positioning. *Localizatio* act of recognizing the environment, that is, assigning consistent labels to different locat *positioning* is the act of computing the coordinates of the robot in the environment. Positi is a task complementary to localization, in the sense that position (e.g., “1.5 meters no of table T”) is often specified in a place-specific coordinate system (“in room 911”). I paper we suggest a method of both localization and positioning using vision alone. A var of the positioning problem, referred to as *repositioning*, involving the return to a pre visited place is also discussed.

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

In the paper we consider a robot that uses a passive sensor, vision, in an indoor environ The environment cannot be changed by the robot to improve its performance; neither beacon nor floor or wall markings are employed. The paper addresses both the localization and positioning problems. Solutions to these problems are presented based on object recogn techniques. The method, based on the linear combinations scheme of [17], represents sce by sets of their 2D images. Localization is achieved by comparing the observed image linear combinations of model views, and the position of the robot is computed by analyz the coefficients of the linear combination that aligns the model to the image. Also, a si “qualitative” solution to the repositioning problem using the linear combinations sch presented.

The rest of the paper is organized as follows. The next section describes the localizat positioning problems and surveys previous solutions. The method of localization and posit using linear combinations of model views is described in Section 3. The method assumes we perspective projection. An iterative scheme to account for perspective distortions is p in Section 4. An analysis of the error resulting from the projection assumption is prese Section 5. Constraints imposed on the motion of the robot as a result of special propert indoor environments can be used to reduce the complexity of the method presented here. Th topic is covered on Section 6. Experimental results follow.

2 The Problem

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

One problem a system for localization and positioning should address is the variabil images due to viewpoint changes. The inexactness of practical systems makes it difficult f robot to return to a specified position on subsequent visits. The visual data available robot between visits varies in accordance with the viewing position of the robot. A local 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 m look different due to changes in the arrangement of the objects, the introduction of new obj and the removal of others. In general, some objects tend to be more static than others. W chairs and books are often moved, tables, closets, and pictures tend to change their po much less, and walls are almost guaranteed to be static. Static cues naturally are more re than mobile ones. Confining the system to static cues, however, may in some cases result failure to recognize the scene due to insufficient cues. The system should therefore atten 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 he computation. Existing systems often try to reduce this cost by using sparse representa and by exploiting contextual information. Sparse representations are introduced in [1 Mataric [10] represents scenes as sequences of landmarks (such as walls, doors, etc.) ex by tracing the boundaries of the scene using a sonar and a compass. Metric information and between the landmarks is not stored. Sarachik [14] recognizes a room by its dimensi which are measured by identifying and locating the top corners of the room using stereo d (obtained from four cameras). In both cases the representation is very sparse, and the sc therefore often ambiguous.

Richer representations are used in [3, 5] where higher success rates are reported. Bra [3] represents the scene by an occupancy table, a 2D bit array which contains a 1 at ev location occupied by some object. The table is constructed by taking stereo pictures co 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.

Engelson *et al.* [5] represent the scene by a set of invariant "signatures". A signat usually composed of low-resolution gray-level or range data obtained by blurring an imag set of signatures taken from different viewpoints are stored. A scene is recognized if th encounters a signature similar to one of the stored signatures.

Systems that use the full information provided by the image (e.g., [6, 12]) usuall on contextual information to avoid scanning all the models in the memory and to reduce t computational cost of comparing a model to the image. The system follows a predetermine

path, so that the identity of each visited location is known in advance, and localization b
a verification problem. Path continuity in many cases is essential, and the so-called "dro
problem is not addressed. The emphasis in these systems is on positioning, which is use
keep the robot on the path. It is typical for these systems (e.g., [1, 6, 12]) to use a
model of the environment.

Onoguchi *et al.* [12], among others, represent the environment by a set of landmarks sele
from pairs of stereo images by a human operator. These landmarks are transformed by an imag
processing program which is designed so as to identify the specific landmark using spe
extraction instructions (such as what features to look for and at what locations). Local
is achieved by applying the extraction procedure specified for the next landmark. Onc
landmark is identified, the position of the robot relative to that landmark is determin
comparing the dimensions of the observed landmark with those of the stored model.

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

3 The Method

The problems of localization and object recognition are similar in many ways. Both probl
require the matching of visual images to stored models, either of the environment or o
observed objects. Both problems face similar difficulties, such as varying illumination con
and changes in appearance due to viewpoint changes. Similar methodologies therefore ca
used for solving both problems.

A particular application of an object recognition scheme, the Linear Combinations (
scheme [17], to the problems of localization and positioning is discussed below. The envir
is represented in this scheme by a small set of views obtained from different viewpoints a
the correspondence between the views. A novel view is recognized by comparing it to li
combinations of the stored views. Positioning is achieved by recovering the position
camera relative to its position in the model views from the coefficients of the aligning
combination. In the rest of this section we review the linear combinations approach and des
its application to both localization and positioning. The section concludes with a sol
the problem of repositioning, that is, the problem of returning to a previously visited
by "locking" into an image acquired in that position.

3.1 Localization

The problem of localization is defined as follows: given P, a 2D image of a place, and $\mathcal{M}$, a s
stored models, find a model $M^i \in \mathcal{M}$ such that P matches M^i. Localization is the recognition
of a place. It can therefore potentially benefit from using an object recognition method
A common approach to handling the problem of recognition from different viewpoints is b
comparing the stored models to the observed environment after the viewpoint is recovered
compensated for. This approach, called *alignment*, is used in a number of studies of ob
recognition [2, 7, 8, 9, 15, 16]. We apply the alignment approach to the problem of localiz
The system described below uses the "Linear Combinations" (LC) scheme, which was suggeste
by Ullman and Basri [17].

We begin with a brief review of the LC scheme. LC is defined as follows. Given an image, we
construct two view vectors from the feature points in the image, one contains the x-coordi
of the points, and the other contains the y-coordinates of the points. An object (in our
the environment) is modeled by a set of such views, where the points in these views are ord
in correspondence. The appearance of a novel view of the object is predicted by apply
linear combinations to the stored views. The predicted appearance is then compared with
actual image, and the object is recognized if the two match. The advantage of this meth
is twofold. First, viewer-centered representations are used rather than object-center
namely, models are composed of 2D views of the observed scene; second, novel appearances a
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 i
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 perspe
projection (orthographic projection and scale). The limitations of this projection m
discussed later in this paper.

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

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

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

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

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

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

or, in other words, $\mathbf{x}'$ and $\mathbf{y}'$ belong to a four-dimensional linear subspace of R^n. (Notice that $\mathbf{z}'$, the vector of depth coordinates of the projected points, also belongs to this subspac fact is used in Section 4 below.) A four-dimensional space is spanned by any four lin independent vectors of the space. Two views of the scene supply four such vectors [13, 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 ther exist coefficients a_1, a_2, a_3, a_4 and b_1, b_2, b_3, b_4 such that

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

(Note that the vector $\mathbf{y}_2$ already depends on the other four vectors.) Since R is a rotatio matrix, the coefficients satisfy the following two quadratic constraints:

$$\begin{aligned} a_1^2 + a_2^2 + a_3^2 - b_1^2 - b_2^2 - b_3^2 &= 2(b_1b_3 - a_1a_3)r_{11} + 2(b_2b_3 - a_2a_3)r_{12} \\ a_1b_1 + a_2b_2 + a_3b_3 + (a_1b_3 + a_3b_1)r_{11} + (a_2b_3 + a_3b_2)r_{12} &= 0 \end{aligned} \tag{5}$$

To derive these constraints the transformation between the two model views should be recove This can be done under weak perspective using a third image. Alternatively, the constra can be ignored, in which case the system would confuse rigid transformations with affine or This usually does not prevent successful localization since generally scenes are fairly from one another.

A LC scheme for the problem of localization is as follows: The environment is model by a set of images with correspondence between the images. For example, a spot can b modeled by two of its corresponding views. The corresponding quadratic constraints may a be stored. Localization is achieved by recovering the linear combination that aligns the to the observed image. The coefficients are determined using four model points and the corresponding image points by solving a linear set of equations. Three points are suffici determine the coefficients if the quadratic constraints are also considered. Additional may be used to reduce the effect of noise.

The LC scheme uses viewer-centered models, that is, representations that are compos of images. It has a number of advantages over methods that build full three-dimensio models to represent the scene. First, by using viewer-centered models that cover relativel transformations we avoid the need to handle occlusions in the scene. If from some viewpo the scene appears different because of occlusions we utilize a new model for these viewpo Second, viewer-centered models are easier to build and to maintain than object-centered The models contain only images and correspondences. By limiting the transformation betwe the model images one can find the correspondence using motion methods. If large portions the environment are changed between visits a new model can be constructed by simply replaci old images with new ones.

One problem with using the LC scheme for localization is due to the weak perspective a proximation. In contrast with the problem of object recognition, where we can generally as that objects are small relative to their distance from the camera, in localization the ment surrounds the robot and perspective distortions cannot be neglected. The limitat of weak perspective modeling are discussed both mathematically and empirically in the two sections. It is shown that in many practical cases weak perspective is sufficient to e accurate localization. The main reason is that the problem of localization does not r accurate measurements in the entire image; it only requires identifying a sufficient numb spots to guarantee accurate naming. If these spots are relatively close to the center image, or if the depth differences they create are relatively small (as in the case of loo a wall when the line of sight is nearly perpendicular to the wall), the perspective dist are relatively small, and the system can identify the scene with high accuracy. Also, related by a translation parallel to the image plane form a linear space even when perspe distortions are large. This case and other simplifications are discussed in Section 6.

By using weak perspective we avoid stability problems that frequently occur in perspec computations. We can therefore compute the alignment coefficients by looking at a relativ narrow field of view. The entire scheme can be viewed as an accumulative process. Rather th acquiring images of the entire scene and comparing them all to a full scene model (as in we recognize the scene image by image, spot by spot, until we accumulate sufficient convinci information that indicates the identity of the place.

When perspective distortions are relatively large and weak perspective is insufficie model the environment, two approaches can be used. One possibility is to construct a la number of models so as to keep the possible changes between the familiar and the novel vie small. Alternatively, an iterative computation can be applied to compensate for these d tions. Such an iterative method is described in Section 4.

3.2 Positioning

Positioning is the problem of recovering the exact position of the robot. This position specified in a fixed coordinate system associated with the environment (i.e., room coordina or it can be associated with some model, in which case location is expressed with respect t position from which the model views were acquired. In this section we discuss an applica of the LC scheme to the positioning problem.

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' relative to the model images is recovered. To recover the absolute position of the robot 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 $t = (t_x, t_y, t_z)$ 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:

$$x' = a_1 x_1 + a_2 y_1 + a_3 x_2 + a_4$$

$$y' = b_1 x_1 + b_2 y_1 + b_3 x_2 + b_4 \tag{6}$$

Substituting for x_2 we obtain

$$\begin{aligned} x' &= a_1 x_1 + a_2 y_1 + a_3 (s r_{11} x_1 + s r_{12} y_1 + s r_{13} z_1 + t_x) + a_4 \\ y' &= b_1 x_1 + b_2 y_1 + b_3 (s r_{11} x_1 + s r_{12} y_1 + s r_{13} z_1 + t_x) + b_4 \end{aligned} \tag{7}$$

and rearranging these equations we obtain

$$\begin{aligned} x' &= (a_1 + a_3 s r_{11}) x_1 + (a_2 + a_3 s r_{12}) y_1 + (a_3 s r_{13}) z_1 + (a_3 t_x + a_4) \\ y' &= (b_1 + b_3 s r_{11}) x_1 + (b_2 + b_3 s r_{12}) y_1 + (b_3 s r_{13}) z_1 + (b_3 t_x + a_4) \end{aligned} \tag{8}$$

Using these equations we can derive all the parameters of the transformation between the m
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

$$\begin{aligned} s_n^2 &= (a_1 + a_3 s r_{11})^2 + (a_2 + a_3 s r_{12})^2 + (a_3 s r_{13})^2 \\ &= a_1^2 + a_2^2 + a_3^2 s^2 + 2 a_3 s (a_1 r_{11} + a_2 r_{12}) \end{aligned} \tag{9}$$

From Equations (8) and (9), by deriving the components of the translation vector, t_n, we
obtain the position of the robot in the image relative to its position in the model views

$$\begin{aligned} \Delta x &= a_3 t_x + a_4 \\ \Delta y &= b_3 t_y + b_4 \\ \Delta z &= f\left(\frac{1}{s_n} - \frac{1}{s}\right) \end{aligned} \tag{10}$$

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

$$\begin{aligned} u_{11} &= \frac{a_1 + a_3 s r_{11}}{s_n} & u_{12} &= \frac{a_2 + a_3 s r_{12}}{s_n} & u_{13} &= \frac{a_3 s r_{13}}{s_n} \\ u_{21} &= \frac{b_1 + a_3 s r_{21}}{s_n} & u_{22} &= \frac{b_2 + a_3 s r_{22}}{s_n} & u_{23} &= \frac{b_3 s r_{23}}{s_n} \end{aligned} \tag{11}$$

As was already mentioned, the position of the robot is computed here relative to the positi
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 elevati
To obtain the relative position the transformation parameters between the model views, P_1
P_2, are required.

3.3 Repositioning

An interesting variant of the positioning problem, referred to as *repositioning*, is def
follows. Given an image, called the *target* image, position yourself in the location fro

this image was observed.[1] One way to solve this problem is to extract the exact position fro
which the target image was obtained and direct the robot to that position. In this secti
are interested in a more qualitative approach. Under this approach position is not comp
Instead, the robot observes the environment and extracts only the direction to the t
location. Unlike the exact approach, the method presented here does not require the rec
of the transformation between the model views.

We assume we are given with a model of the environment together with a target image. The robot is allowed to take new images as it is moving towards the target. We assume
horizontally moving platform (In other words, we assume three degrees of freedom rather
six; the robot is allowed to rotate around the vertical axis and translate horizontal
validity of this constraint is discussed in Section 6.) Below we give a simple computati
determines a path which terminates in the target location. At each time step the robot acq
a new image and aligns it with the model. By comparing the alignment coefficients with th
coefficients for the target image the robot determines its next step. The algorithm is di
into two stages. In the first stage the robot fixates on one identifiable point and moves a
a circular path around the fixation point until the line of sight to this point coincid
the line of sight to the corresponding point in the target image. In the second stage the
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. 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

$$\begin{aligned} x_t &= a_1 x_1 + a_3 x_2 + a_4 \\ y_t &= b_2 y_1 \end{aligned} \tag{12}$$

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

$$\begin{aligned} a_1 &= \frac{s_t \sin(\alpha - \theta)}{\sin\alpha} \\ a_3 &= \frac{s_t \sin\theta}{s \sin\alpha} \\ a_4 &= t_t - \frac{t_x s_t \sin\theta}{s \sin\alpha} \\ b_2 &= s_t \end{aligned} \tag{13}$$

(The derivation is given in the Appendix.)

At every time step the robot acquires an image and aligns it with the above model. Assu
that image P_p is obtained as a result of a rotation by an angle ϕ, translation t_p and scale s_p

[1] This problem can be considered as a variant of the homing problem. A discussion of the general homing problem with a "signature-based" solution can be found in [11].

The position of a point, (x_p, y_p) is expressed by

$$\begin{aligned} x_p &= c_1 x_1 + c_3 x_2 + c_4 \\ y_p &= d_2 y_1 \end{aligned} \tag{14}$$

where the coefficients are given by

$$\begin{aligned} c_1 &= \frac{s_p \sin(\alpha - \phi)}{\sin\alpha} \\ c_3 &= \frac{s_p \sin\phi}{s \sin\alpha} \\ c_4 &= t_p - \frac{t_x s_p \sin\phi}{s \sin\alpha} \\ d_2 &= s_p \end{aligned} \tag{15}$$

The step performed by the robot is determined by

$$\delta = \frac{c_1}{c_3} - \frac{a_1}{a_3} \tag{16}$$

That is,

$$\delta = \frac{s \sin(\alpha - \phi)}{\sin\phi} - \frac{s \sin(\alpha - \theta)}{\sin\theta} = s \sin\alpha(\cot\phi - \cot\theta) \tag{17}$$

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

A new image P_n is now acquired, and the fixated point is located in this image. Denot its new position by x_n. Since the motion is parallel to the image plane the depth values of t 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 deduce the equation

$$x_p = x_n \cos\beta + \sin\beta \tag{18}$$

This equation has two solutions. We chose the one that counters the translation (namel translation is to the right, the camera should rotate to the left), and that keeps the a 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 coin with that of the target image, and $\phi = \theta$) it should now advance forward or retreat backwa to adjust its position along the line of sight. Several measures can be used to determi direction of motion; one example is the term c_1/a_1 which satisfies

$$\frac{c_1}{a_1} = \frac{s_p}{s_t} \tag{19}$$

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

4 Handling Perspective Distortions

The linear combination scheme presented above accurately handles changes in viewpoint ass ing the images are obtained under weak perspective projection. Error analysis and experim results demonstrate that in many practical cases this assumption is valid. In cases wher spective distortions are too large to be handled by a weak perspective approximation, mat between the model and the image can be facilitated in two ways. One possibility is to av cases of large perspective distortion by augmenting the library of stored models with addi models. In a relatively dense library there usually exists a model that is related to th by a sufficiently small transformation avoiding such distortions. The second alternative improve the match between the model and the image using an iterative process. In this sect we consider the second option.

The suggested iterative process is based on a Taylor expansion of the perspective co nates. As described below, this expansion results in a polynomial consisting of term of which can be approximated by linear combinations of views. The first term of this ser represents the orthographic approximation. The process resembles a method of matching points with 2D points described recently by DeMenthon and Davis [4]. In this case, howev the method is applied to 2D models rather than 3D ones. In our application the 3D coordinat of the model points are not provided; instead they are approximated from the model views.

An image point $(x, y) = (fX/Z, fY/Z)$ is the projection of some object point, $(X, Y,$ the image, where f denotes the focal length. Consider the following Taylor expansion of around some depth value Z_0

$$
\begin{aligned}
\frac{1}{Z} &= \sum_{k=0}^{\infty} \frac{f^{(k)}(Z_0)}{k!}(Z - Z_0)^k \\
&= \frac{1}{Z_0} + \sum_{k=1}^{\infty} \frac{(-1)^k}{(k-1)!} \frac{(Z - Z_0)^k}{Z_0^{k+1}} \\
&= \frac{1}{Z_0}\left[1 + \sum_{k=1}^{\infty} \frac{(-1)^k}{(k-1)!}\left(\frac{Z - Z_0}{Z_0}\right)^k\right]
\end{aligned}
\tag{20}
$$

The Taylor series describing the position of a point x is therefore given by

$$
x = \frac{fX}{Z} = \frac{fX}{Z_0}\left[1 + \sum_{k=1}^{\infty} \frac{(-1)^k}{(k-1)!}\left(\frac{Z - Z_0}{Z_0}\right)^k\right]
\tag{21}
$$

Notice that the zero term contains the orthographic approximation for x. Denote by $\Delta^{(k)}$ the kth term of the series:

$$
\Delta^{(k)} = \frac{fX}{Z_0}\frac{(-1)^k}{(k-1)!}\left(\frac{Z - Z_0}{Z_0}\right)^k
\tag{22}
$$

A recursive definition of the above series is given below.

Initialization:

$$x^{(0)} = \Delta^{(0)} = \frac{fX}{Z_0}$$

Iterative step:

$$\begin{aligned} \Delta^{(k)} &= -\frac{Z - Z_0}{(k-1)Z_0}\Delta^{(k-1)} \\ x^{(k)} &= x^{(k-1)} + \Delta^{(k)} \end{aligned}$$

where $x^{(k)}$ represents the kth order approximation for x, and $\Delta^{(k)}$ represents the highest order term in $x^{(k)}$.

According to the orthographic approximation both X and Z can be expressed as linear com- binations of the model views (Eq. (4)). We therefore apply the above procedure, approximat X and Z at every step using the linear combination that best aligns the model points with image points. The general idea is therefore the following. First, we estimate $x^{(0)}$ and $\Delta^{(0)}$ by solving the orthographic case. Then at each step of the iteration we improve the estimat seeking the linear combination that best estimates the factor

$$-\frac{Z - Z_0}{(k-1)Z_0} \approx \frac{x - x^{(k-1)}}{\Delta^{(k-1)}} \tag{23}$$

Denote by $\mathbf{x} \in \mathcal{R}$ the vector of image point coordinates, and denote by

$$P = [\mathbf{x}_1, \mathbf{y}_1, \mathbf{x}_2, \mathbf{1}] \tag{24}$$

an $n \times 4$ matrix containing the position of the points in the two model images. Denote $P^+ = (P^T P)^{-1} P^T$ the pseudo-inverse of P (we assume P is overdetermined). Also denote by $\mathbf{a}^{(k)}$ the coefficients computed for the kth step. $P\mathbf{a}^{(k)}$ represents the linear combination computed at that step to approximate the X or the Z values. Since at every step f, Z_0 and k are constant they can be merged into the linear combination. Denote by $\mathbf{x}^{(k)}$ and by $\Delta\mathbf{x}^{(k)}$ the vectors of computed values of x and Δ at the kth step. An iterative procedure to align a mo to the image is described below.

Initialization:
Solve the orthographic approximation, namely

$$\begin{aligned} \mathbf{a}^{(0)} &= P^+\mathbf{x} \\ \mathbf{x}^{(0)} = \mathbf{\Delta}^{(0)} &= P\mathbf{a}^{(0)} \end{aligned}$$

Iterative step:

$$\begin{aligned} \mathbf{q}^{(k)} &= (\mathbf{x} - \mathbf{x}^{(k-1)}) \div \Delta^{(k-1)} \\ \mathbf{a}^{(k)} &= P^+\mathbf{q}^{(k)} \\ \Delta^{(k)} &= (P\mathbf{a}^{(k)}) \otimes \mathbf{\Delta}^{(k-1)} \\ \mathbf{x}^{(k)} &= \mathbf{x}^{(k-1)} + \mathbf{\Delta}^{(k)} \end{aligned}$$

where the vector operations $\otimes$ and $\div$ are defined as

$$
\begin{aligned}
\mathbf{u} \otimes \mathbf{v} &= (u_1 v_1, \ldots, u_n v_n) \\
\mathbf{u} \div \mathbf{v} &= (\frac{u_1}{v_1}, \ldots, \frac{u_n}{v_n})
\end{aligned}
$$

5 Projection Model – Error Analysis

In this section we estimate the error obtained by using the linear combination method. method assumes a weak perspective projection model. We compare this assumption with the more accurate perspective projection model.

A point (X, Y, Z) is projected under the perspective model to $(x, y) = (fX/Z, fY/Z)$ i image, where f denotes the focal length. Under our weak perspective model the same poi is approximated by $(\hat{x}, \hat{y}) = (sX, sY)$ where s is a scaling factor. The best estimate for scaling factor, is given by $s_0 = f/Z_0$, where Z_0 is the average depth of the observed environment. Denote the error by

$$E = |\hat{x} - x| \tag{25}$$

The error is expressed by

$$E = \left| fX(\frac{1}{Z_0} - \frac{1}{Z}) \right| \tag{26}$$

Changing to image coordinates

$$E = \left| xZ(\frac{1}{Z_0} - \frac{1}{Z}) \right| \tag{27}$$

or

$$E = |x| \left| \frac{Z}{Z_0} - 1 \right| \tag{28}$$

The error is small when the measured feature is close the optical axis, or when our esti for the depth, Z_0, is close to the real depth, Z. This supports the basic intuition tha images with low depth variance and for fixated regions (regions near the center of the ima the obtained perspective distortions are relatively small, and the system can therefore the scene with high accuracy. Figures 1 and 2 show the depth ratio Z/Z_0 as a function of x for $\epsilon = 10$ and 20 pixels, and Table 5 shows a number of examples for this function. The allowe depth variance, Z_0/Z, is computed as a function of x and the tolerated error, ϵ. For exampl a 10 pixel error tolerated in a field of view of up to ± 50 pixels is equivalent to allowin variations of 20%. From this discussion it is apparent that when a model is aligned to the the results of this alignment should be judged differently at different points of the image farther away a point is from the center the more discrepancy should be tolerated between prediction and the actual image. A five pixel error at position $x = 50$ is equivalent to a 10 error at position $x = 100$.

So far we have considered the discrepancies between the weak perspective and the persp tive projections of points. The accuracy of the LC scheme depends on the validity of the

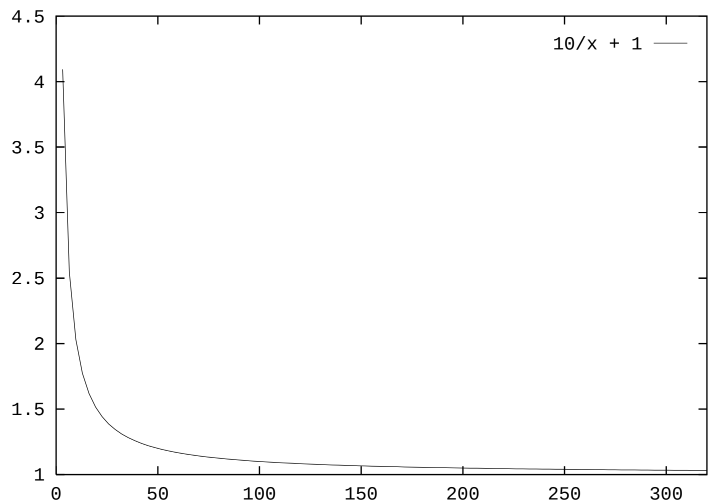

Figure 1: $\frac{Z}{Z_0}$ as a function of x for $\epsilon = 10$ pixels.

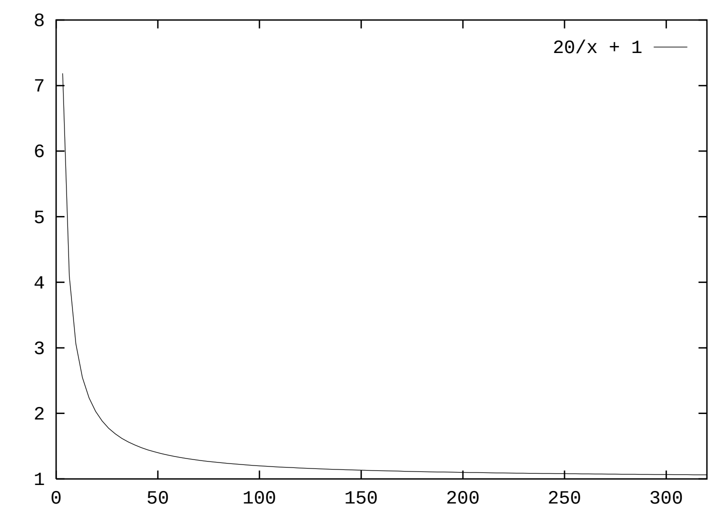

Figure 2: $\frac{Z}{Z_0}$ as a function of x for $\epsilon = 20$ pixels.

$x \setminus \epsilon$	5	10	15	20
25	1.2	1.4	1.6	1.8
50	1.1	1.2	1.3	1.4
75	1.07	1.13	1.2	1.27
100	1.05	1.1	1.15	1.2

Table 1: Allowed depth ratios, $\frac{Z}{Z_0}$, as a function of x (half the width of the field considered) and the error allowed (ϵ, in pixels).

perspective projection both in the model views and for the incoming image. In the rest of section we develop an error term for the LC scheme assuming that both the model views and the incoming image are obtained by perspective projection.

The error obtained by using the LC scheme is given by

$$E = |x - a x_1 - b y_1 - c x_2 - d| \tag{29}$$

Since the scheme accurately predicts the appearances of points under weak perspective projection, it satisfies

$$\hat{x} = a\hat{x}_1 - b_1\hat{y}_1 - c\hat{x}_2 - d \tag{30}$$

where accented letters represent orthographic approximations. Assume that in the two model pictures the depth ratios are roughly equal:

$$\frac{Z_0^M}{Z^M} = \frac{Z_{01}}{Z_1} \approx \frac{Z_{02}}{Z_2} \tag{31}$$

(This condition is satisfied, for example, when between the two model images the camera only translates along the image plane.) Using the fact that

$$x = \frac{fX}{Z} = \frac{fX}{Z_0}\frac{Z_0}{Z} = \hat{x}\frac{Z_0}{Z} \tag{32}$$

we obtain

$$\begin{aligned}
E &= |x - a x_1 - b y_1 - c x_2 - d| \\
&\approx \left|\hat{x}\frac{Z_0}{Z} - a\hat{x}_1\frac{Z_0^M}{Z^M} - b_1\hat{y}_1\frac{Z_0^M}{Z^M} - c\hat{x}_2\frac{Z_0^M}{Z^M} - d\right| \\
&= \left|\hat{x}\frac{Z_0}{Z} - (a_1\hat{x}_1 - b_1\hat{y}_1 - c_2\hat{x}_2)\frac{Z_0^M}{Z^M} - d\right| \\
&= \left|\hat{x}\frac{Z_0}{Z} - (\hat{x} - d)\frac{Z_0^M}{Z^M} - d\right|
\end{aligned} \tag{33}$$

$$
\begin{aligned}
&= \left| \hat{x}\left(\frac{Z_0}{Z} - \frac{Z_0^M}{Z^M}\right) - d\left(\frac{Z_0^M}{Z^M} - 1\right) \right| \\
&\leq |\hat{x}| \left| \frac{Z_0}{Z} - \frac{Z_0^M}{Z^M} \right| + |d| \left| \frac{Z_0^M}{Z^M} - 1 \right|
\end{aligned}
$$

The error therefore depends on two terms. The first gets smaller as the image points get cl to the center of the frame and as the difference between the depth ratios of the model and image gets smaller. The second gets smaller as the translation component gets smaller ar the model gets close to orthographic.

Following this analysis, weak perspctive can be used as a projection model when the de variations in the scene are relatively low and when the system concentrates on the center of the image. We conclude that, by fixating on distinguished parts of the environment, linear combinations scheme can be used for localization and positioning.

6 Imposing Constraints

Localization and positioning require a large memory and a great deal of on-line computat A large number of models must be stored to enable the robot to navigate and manipulate in relatively large and complicated environments. The computational cost of model-ir comparison is high, and if context (such as path history) is not available the number of re comparisons may get very large. To reduce this computational cost a number of constraints r be employed. These constraints take advantage of the structure of the robot, the propert indoor environments, and the natural properties of the navigation task. This section exa some of these constraints.

One thing a system may attempt to do is to build the set of models so as to reduce th effect of perspective distortions in order to avoid performing iterative computations. of the environment obtained when the system looks relatively deep into the scene usua satisfy this condition. When perspective distortions are large the system may consider mo subsets of views related by a translation parallel to the image plane (perpendicular to t of sight). In this case the depth values of the points are roughly equal across all considered, and it can be shown that novel views can be expressed by linear combinations two model views even in the presence of large perspective distortions. This becomes appa from the following derivation. Let (X_i, Y_i, Z_i), $1 \leq i \leq n$ be a point projected in the image to $(x_i, y_i) = (fX_i/Z_i, fY_i/Z_i)$, and let (x'_i, y'_i) be the projected point after applying a rigid transformation. Assuming that $Z'_i = Z_i$ we obtain

$$
\begin{aligned}
Z_i x'_i &= r_{11} X_i + r_{12} Y_i + r_{13} Z_i + t_x \\
Z_i y'_i &= r_{21} X_i + r_{22} Y_i + r_{23} Z_i + t_y
\end{aligned}
\qquad (34)
$$

Dividing by Z_i we obtain

$$
x'_i = r_{11} x_i + r_{12} y_i + r_{13} + t_x \frac{1}{Z_i}
$$

$$y_i' = r_{21}x_i + r_{22}y_i + r_{23} + t_y \frac{1}{Z_i} \qquad (35)$$

Rewriting this in vector equation form gives

$$\begin{aligned} \mathbf{x}' &= r_{11}\mathbf{x} + r_{12}\mathbf{y} + r_{13}\mathbf{1} + t_x\mathbf{z}^{-1} \\ \mathbf{y}' &= r_{21}\mathbf{x} + r_{22}\mathbf{y} + r_{23}\mathbf{1} + t_y\mathbf{z}^{-1} \end{aligned} \qquad (36)$$

where $\mathbf{x}$, $\mathbf{y}$, $\mathbf{x}'$, and $\mathbf{y}'$ are the vectors of x_i, y_i, x_i', and y_i' values respectively, $\mathbf{1}$ is a vector of all 1s, and $\mathbf{z}^{-1}$ is a vector of $1/Z_i$ values. Consequently, as in the weak perspective case, novel views obtained by a translation parallel to the image plane can be expressed by li combinations of four vectors.

An indoor environment usually provides the robot with a flat, horizontal support. Con quently, the motion of the camera is often constrained to rotation about the vertical (Y and to translation in the XZ-plane. Such motion has only three degrees of freedom instea the six degrees of freedom in the general case. Under this constraint fewer corresponden required to align the model with the image. For example, in Eq. (4) (above) the coeffic $a_2 = b_1 = b_3 = b_4 = 0$. Three points rather than four are required to determine the coefficient by solving a linear system. Two, rather than three, are required if the quadratic constrai also considered. Another advantage to considering only horizontal motion is the fact tha motion constrains the possible epipolar lines between images. This fact can be used to the task of correspondence seeking.

Objects in indoor environments sometimes appear in roughly planar settings. In partic the relatively static objects tend to be located along walls. Such objects include w shelves, pictures, closets and tables. When the assumption of orthographic projection i (for example, when the robot is relatively distant from the wall, or when the line of si roughly perpendicular to the wall) the transformation between any two views can be descri by a 2D affine transformation. The dimension of the space of views of the scene is then reduc to three (rather than four), and Eq. (4) becomes

$$\begin{aligned} \mathbf{x}' &= a_1\mathbf{x}_1 + a_2\mathbf{y}_1 + a_4\mathbf{1} \\ \mathbf{y}' &= b_1\mathbf{x}_1 + b_2\mathbf{y}_1 + b_4\mathbf{1} \end{aligned} \qquad (37)$$

($a_3 = b_3 = 0$.) Only one view is therefore sufficient to model the scene.

Most office-like indoor environments are composed of rooms connected by corridors. Nav gating in such an environment involves maneuvering through the corridors, entering and exi the rooms. Not all points in such an environment are equally important. Junctions, places the robot faces a number of options for changing its direction, are more important than places for navigation. In an indoor environment these places include the thresholds of and the beginnings and ends of corridors. A navigation system would therefore tend to s more models for these points than for others.

One important property shared by many junctions is that they are confined to relative small areas. Consider for example the threshold of a room. It is a relatively narrow

that separates the room from the adjacent corridor. When a robot is about to enter a roo
a common behavior includes stepping through the door, looking into the room, and identify
it before a decision is made to enter the room or to avoid it. The set of interesting imag
this task includes the set of views of the room from its entrance. Provided that threshol
narrow these views are related to each other almost exclusively by rotation around the ve
axis. Under perspective projection, such a rotation is relatively easy to recover. The
of points in novel views can be recovered from one model view only. This is apparent fr
the following derivation. Consider a point $p = (X, Y, Z)$. Its position in a model view is
by $(x, y) = (fX/Z, fY/Z)$. Now, consider another view obtained by a rotation R around t
camera. The location of p in the new view is given by (assuming $f = 1$)

$$(x', y') = \left(\frac{r_{11}X + r_{12}Y + r_{13}Z}{r_{31}X + r_{32}Y + r_{33}Z}, \frac{r_{21}X + r_{22}Y + r_{23}Z}{r_{31}X + r_{32}Y + r_{33}Z}\right) \quad (38)$$

implying that

$$(x', y') = \left(\frac{r_{11}x + r_{12}y + r_{13}}{r_{31}x + r_{32}y + r_{33}}, \frac{r_{21}x + r_{22}y + r_{23}}{r_{31}x + r_{32}y + r_{33}}\right) \quad (39)$$

Depth is therefore not a factor in determining the relation between the views. Eq. (39) be
even simpler if only rotations about the Y-axis are considered:

$$(x', y') = \left(\frac{x\cos\alpha + \sin\alpha}{-x\sin\alpha + \cos\alpha}, \frac{y}{-x\sin\alpha + \cos\alpha}\right) \quad (40)$$

where α is the angle of rotation. In this case α can be recovered merely from a single c
spondence.

7 Experiments

The LC method was implemented and applied to images taken in an indoor environment.
Images of two offices, A and B, that have similar structures were taken using a Panasonic came
with a focal length of 700 pixels. Semi-static objects, such as heavy furniture and pictur
used to distinguish between the offices. Figure 3 shows two model views of office A. The views
were taken at a distance of about 4m from the wall. Correspondences were picked manuall
The results of aligning the model views to images of the two offices are presented in Figur
The left image contains an overlay of a predicted image (the thick white lines), construc
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 f
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,
match any features in the image.

Figure 5 shows the matching of the model of office A with an image of the same office obtained by a relatively large motion forward (about 2m) and to the side (about 1.5m). Altho

Figure 3: Two model views of office A.

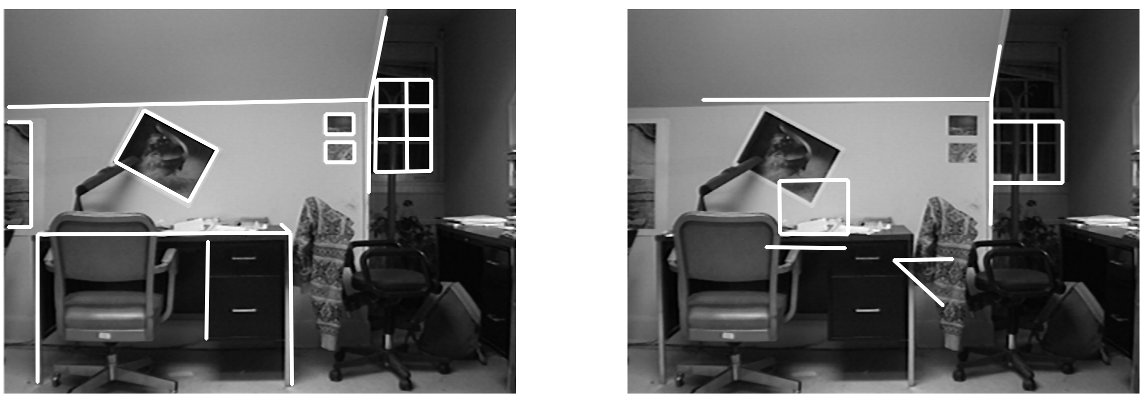

Figure 4: Matching a model of office A to an image of office A (left), and matching a model of office B to the same image (right).

Figure 5: Matching a model of office A to an image of the same office obtained by a relatively large motion forward and to the right.

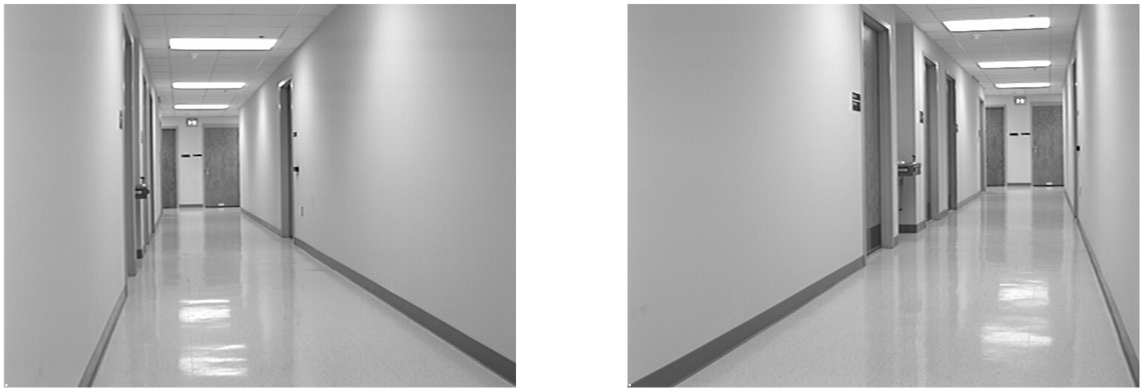

Figure 6: Two model views of a corridor.

Figure 7: Matching the corridor model with two images of the corridor. The right image w
obtained by a relatively large motion forward (about half of the corridor length) and t
right.

the distances are relatively short most perspective distortions are negligible, and a goo
between the model and the image is obtained.

Another set of images was taken in a corridor. Here, because of the deep structure
the corridor, perspective distortions are noticeable. Nevertheless, the alignment re
demonstrate an accurate match in large portions of the image. Figure 6 shows two model vie
of the corridor. Figure 7 (left) shows an overlay of a linear combination of the model
with an image of the corridor. It can be seen that the parts that are relatively distant
perfectly. Figure 7 (right) shows the matching of the corridor model with an image obtaine
a relatively large motion (about half of the corridor length). Because of perspective dis
the relatively near features no longer align (e.g., the near door edges). The relatively
however, still match.

The next experiment shows the application of the iterative process presented in Secti

in cases where large perspective distortion were noticeable. Figure 8 shows two model v
and Figure 9 shows the results of matching a linear combination of the model views to
image of the same office. In this case, because the image was taken from a relatively cl
distance, perspective distortions cannot be neglected. The effects of perspective distort
be noticed on the right corner of the board, and on the edges of the hanger on the top ri
Perspective effects were reduced by using the iterative process. The results of applyi
procedure after one and three iterations are shown in Figure 10.

The experimental results demonstrate that the LC method achieves accurate localizatio
many cases, and that when the method fails because of large perspective distortions an ite
computation can be used to improve the quality of the match.

8 Conclusions

A method of localization and positioning in an indoor environment was presented. The meth
is based on representing the scene as a set of 2D views and predicting the appearance of n
views by linear combinations of the model views. The method accurately approximates th
appearances of scenes under weak perspective projection. Analysis of this projection
as experimental results demonstrate that in many cases this approximation is sufficient
accurately describe the scene. When the weak perspective approximation is invalid, eit
larger number of models can be acquired or an iterative solution can be employed to acco
for the perspective distortions.

The method presented in this paper has several advantages over existing methods. It u
relatively rich representations; the representations are 2D rather than 3D, and localiza
be done from a single 2D view only. The same basic method is used in both the localizati
and positioning problems, and a simple algorithm for repositioning is derived from this me
Future work includes handling the problem of acquisition and maintenance of models, devel
ing efficient and robust algorithms for solving the correspondence problem, and building m
using visual input.

Appendix

In this appendix we derive the explicit values of the coefficients of the linear combinations
case of horizontal motion. Consider a point $p = (x, y, z)$ that is projected by weak persp
to three images P_1, P_2, and P', P_2 is obtained from P_1 by a rotation about the Y-axis by an angle α, translation t_m and scale factor s_m, and P' is obtained from P_1 rotation about the Y-axis by an angle θ, translation t_p and scale s_p. The position of p in the three images is given by

$$(x_1, y_1) = (x, y)$$

Figure 8: Two model views of office C.

Figure 9: Matching the model to an image obtained by a relatively large motion. Perspect
distortions can be seen in the table, the board, and the hanger at the upper right.

Figure 10: The results of applying the iterative process to reduce perspective distorti
one (left) and three (right) iterations.

$$
\begin{aligned}
(x_2, y_2) &= (s_m x \cos\alpha + s_m z \sin\alpha + t_m,\ s_m y) \\
(x', y') &= (s_p x \cos\theta + s_p z \sin\theta + t_p,\ s_p y)
\end{aligned}
$$

The point (x', y') can be expressed by a linear combination of the first two points:

$$
\begin{aligned}
x' &= a_1 x_1 + a_2 x_2 + a_3 \\
y' &= b\, y_1
\end{aligned}
$$

Rewriting these equations we get

$$
\begin{aligned}
s_p x \cos\theta + s_p z \sin\theta + t_p &= a_1 x + a_2 (s_m x \cos\alpha + s_m z \sin\alpha + t_m) + a_3 \\
s_p y &= b\, y
\end{aligned}
$$

Equating the values for the coefficients in both sides of these equations we obtain

$$
\begin{aligned}
s_p \cos\theta &= a_1 + a_2 s_m \cos\alpha \\
s_p \sin\theta &= a_2 s_m \sin\alpha \\
t_p &= a_2 t_m + a_3 \\
s_p &= b
\end{aligned}
$$

and the coefficients are therefore given by

$$
\begin{aligned}
a_1 &= \frac{s_p \sin(\alpha - \theta)}{\sin\alpha} \\
a_3 &= \frac{s_p \sin\theta}{s_m \sin\alpha} \\
a_4 &= t_p - \frac{t_m s_p \sin\theta}{s_m \sin\alpha} \\
b &= s_p
\end{aligned}
$$

References

[1] N. Ayache and O. D. Faugeras. Maintaining representations of the environment of a mobi robot, *IEEE Trans. on Robotics and Automation*, Vol. 5, pp. 804–819, 1989.

[2] R. Basri and S. Ullman. The alignment of objects with smooth surfaces. *Proc. 2nd I Conf. on Computer Vision*, Tarpon Springs, FL, pp. 482–488, 1988.

[3] D. J. Braunegg. Marvel—A system for recognizing world locations with stereo vision *TR-1229, MIT*, 1990.

[4] D. F. DeMenthon and L. S. Davis. Model-based object pose in 25 lines of code. *Proc. 2 European Conf. on Computer Vision*, Genova, Italy, 1992.

[5] S. P. Engelson and D. V. McDermott. Image signatures for place recognition and ma construction. *Proc. SPIE Symposium on Intelligent Robotic Systems*, Boston, MA, 1991.

[6] 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.

[7] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitti with application to image analysis and automated cartography. *Communications of the ACM*, Vol. 24, pp. 381–395, 1981.

[8] D. P. Huttenlocher and S. Ullman. Object recognition using alignment. *Proc. 1st Int. C on Computer Vision*, London, UK, pp. 102–111, 1987.

[9] D. G. Lowe. Three-dimensional object recognition from single two-dimensional ima *Robotics Research Technical Report 202, Courant Institute of Math. Sciences, New York University*, 1985.

[10] M. J. Mataric. Environment learning using a distributed representation. *Proc. Int. on Robotics and Automation*, Cincinnati, OH, 1990.

[11] R. N. Nelson. Visual homing using an associative memory. *DARPA Image Understanding Workshop*, pp. 245-262, 1989.

[12] K. Onoguchi, M. Watanabe, Y. Okamoto, Y. Kuno, and H. Asada. A visual navigation system using a multi information local map. *Proc. Int. Conf. on Robotics and Automation*, Cincinnati, OH, pp. 767–774, 1990.

[13] T. Poggio. 3D object recognition: on a result by Basri and Ullman. *Technical Repo 9005-03, IRST*, Povo, Italy, 1990.

[14] K. B. Sarachik. Visual navigation: constructing and utilizing simple maps of an i environment. *AI-TR-1113, MIT*, 1989.

[15] 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.

[16] S. Ullman. Aligning pictorial descriptions: an approach to object recognition. *Co* Vol. 32, pp. 193–254, 1989.

[17] S. Ullman and R. Basri. Recognition by linear combinations of models. *IEEE Trans. o Pattern Analysis and Machine Intelligence,* Vol. 13, pp. 992–1006, 1991.