Recognizing Large Isolated 3-D Objects through Next View Planning using Inner Camera Invariants

Sumantra Dutta Roy, Santanu Chaudhury, and Subhashis Banerjee

Abstract— **Most model-based 3-D object recognition systems use information from a single view of an object. However, a single view may not contain sufficient features to recognize it unambiguously. Further, two objects may have all views in common with respect to a given feature set, and may be distinguished only through a sequence of views. A further complication arises when in an image, we do not have a complete view of an object. This paper presents a new on-line scheme for the recognition and pose estimation of a *large* isolated 3-D object, which may not entirely fit in a camera's field of view. We consider an uncalibrated projective camera, and consider the case when the internal parameters of the camera may be varied either unintentionally, or on purpose. The scheme uses a probabilistic reasoning framework for recognition and next view planning. We show results of successful recognition and pose estimation even in cases of a high degree of interpretation ambiguity associated with the initial view.**

Index Terms— **Active 3-D Object Recognition, Next View Planning, Pose Estimation, Inner Camera Invariants**

I. Introduction

IN this paper, we present a new next view planning-based recognition and pose estimation scheme for an isolated large 3-D object. Our approach can handle the situation when a large 3-D object does not fit into a camera's field of view. Fig. 1(a) shows an image of a portion of a building obtained using an *active camera* (one whose parameters can be changed purposively *e.g.*, as in Fig. 2). Such a view could have come from any of the three models, in Fig. 1(b), (c) and (d), respectively. Further, even if the identity of the object were known, the same view could occur at more than one place in the object – it is not possible to know the exact pose of the camera with respect to the object from one view alone.

We present a new reactive object recognition scheme which uses a hierarchical part-based knowledge representation scheme, and a probabilistic framework for both recognition and planning. The planning scheme is independent of the particular nature of a 2-D/3-D part, and the method used to detect it. A novel feature of our work is the use of **Inner Camera Invariants** [1], [2], [3] for pose estimation – image-computable functions which are independent of the internal parameters of a camera.

Most model-based object recognition systems use information from a single view of an object [4], [5], [6]. However,

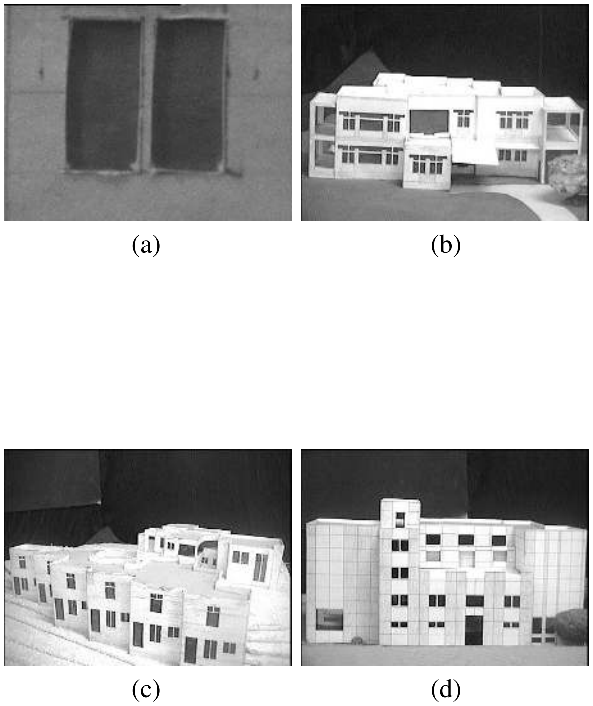
(a) (b)
(c) (d)

Fig. 1. (a) The given view of an object: only a portion of it is visible. This could have come from any of the models: (b), (c) and (d)

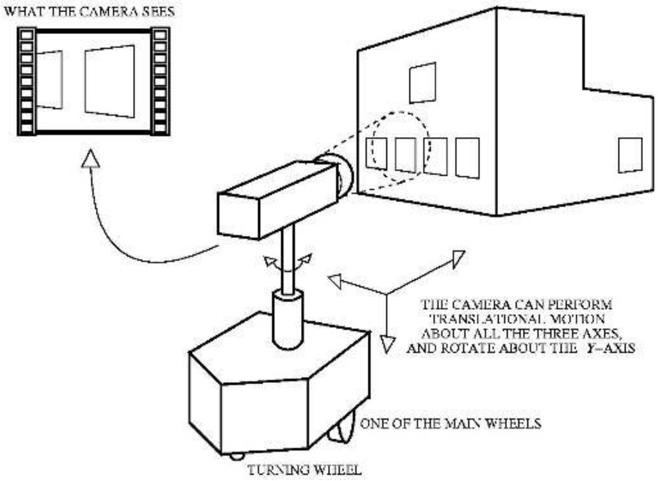
WHAT THE CAMERA SEES
THE CAMERA CAN PERFORM TRANSLATIONAL MOTION ABOUT ALL THE THREE AXES, AND ROTATE ABOUT THE Y–AXIS
ONE OF THE MAIN WHEELS
TURNING WHEEL

Fig. 2. A robot with an attached camera, observing a building. The entire object does not fit in the camera's field of view. Not only is the identity of the object unknown, the robot also does not know its pose with respect to the object. This example shows 4 degrees of freedom (DOF) between the object and the camera.

S. Dutta Roy is with the Dept. of Elec. Engg., I.I.T. Bombay, Powai, Mumbai - 400 076. Email: sumantra@ee.iitb.ac.in

S. Chaudhury is with the Dept. of Elec. Engg., I.I.T. Delhi, Hauz Khas, New Delhi - 110 016. Email: santanuc@{ee, cse}.iitd.ac.in

S. Banerjee is with the Dept. of Comp. Sc. and Engg., I.I.T. Delhi, Hauz Khas, New Delhi - 110 016. Email: suban@cse.iitd.ac.in

a single view may not contain sufficient features to recognize an object unambiguously. In fact, two objects may have all views in common with respect to a feature set, and may be distinguished only through a sequence of views [7]. Further, in recognizing 3-D objects from a single view, recognition systems often use complex feature sets [5], [6]. For object recognition, one needs an effective representation of properties (geometric, photometric, etc.) of objects from images which are invariant to the view point, and should be computable from image information. Invariants may be colour-based (e.g., [8]), photometric (e.g., [9]) or geometric (e.g., [6], [10], [11], [12]). Although Burns, Weiss and Riseman prove a theorem in [13] that invariants cannot be computed for a set of 3-D points in general position, from a single image, geometric invariants have been proposed for a constrained set of 3-D points [6], [10], [11], [12]. Such approaches use the inherent symmetry present in an object, or a particular configuration of objects to compute invariants for recognition (*e.g.*, rotationally symmetric objects, translationally repeated objects, canal surfaces, quadrics, etc.) However, we often need to recognize 3-D objects which because of their inherent asymmetry, cannot be completely characterized by an invariant computed from a single view. Beis and Lowe [14] propose a kd-tree-based alternate indexing strategy, as against using invariants. Lowe's early work *e.g.,* [15] focuses on the use of perceptual grouping for locating features 'invariant' over a wide range of viewpoints. Most of the work is limited to specific geometric information alone. However, the basic premise in all the above methods is in using information from a single image. In many cases, it may be possible to achieve the same, incurring less error and smaller processing cost, using a simpler feature set and suitably planned multiple observations [16], [17]. The purposive control over the parameters of a sensor (both internal as well as external) characterizes an *Active Sensor*. Papers on Active Vision and Sensor Planning include the works of Aloimonos *et al.* [18], Bajcsy [19], Ballard and Brown [20], Tarabanis, Allen and Tsai [21], and the authors' own work [22].

Grimson [23] proposes sensing strategies for disambiguating between multiple objects in known poses. Madsen and Christensen [24] propose a method for viewpoint planning, but for polyhedral objects alone. Further, the authors assume a knowledge of camera internal parameters. Examples of active object recognition systems include those of Maver and Bajcsy [25], Hutchinson and Kak [26], Gremban and Ikeuchi [7], Dickinson *et al.* [27], Callari and Ferrie [28], Borotschnig *et al.* [29], Schiele and Crowley [30], and the authors' own earlier work [16], [17]. We compare different active 3-D object recognition systems on the basis of the following properties:

1) *Features used for modeling and view recognition*
While many approaches such as those of Hutchinson and Kak [26] and Liu and Tsai [31] use geometric features, appearance-based methods such as that of Borotschnig *et al.* [29] use pixel information from an entire image. Dickinson *et al.* [32], [27] use volumetric primitives, which are associated with a high feature extraction cost. The same is true for the super-ellipsoids of Callari and Ferrie [28]. The scheme of Gremban and Ikeuchi [7] and our earlier work [16], [17] can work with any set of features.
2) *The system setup and viewing geometry*
Most multiple view-based approaches using geometric features, implicitly or otherwise, assume the camera model to be orthographic. Most experimentation is with a single (rotational) degree of freedom (DOF, hereafter) between the object and the camera.
3) *Efficient representation of domain knowledge*
Dickinson *et al.* [32], [27] use a hierarchical representation scheme based on volumetric primitives. Borotschnig *et al.* [29] use a parametric eigenspace-based representation, which is associated with a high storage and processing cost. In our earlier work[16], [17], the hierarchy itself enforces different constraints to prune the set of possible hypotheses. Due to the non-hierarchical nature of Hutchinson and Kak's system [26], many redundant hypotheses are proposed, which have to be later removed through consistency checks.
4) *Speed and efficiency of algorithms for both hypothesis generation and next view planning*
In Hutchinson and Kak's system [26], the polynomial-time formulation overcomes the exponential time complexity associated with assigning beliefs to all possible hypotheses. However, their system still has the overhead of intersection computation in creating common frames of discernment. Consistency checks have to be used to remove the many redundant hypotheses produced earlier. Though Dickinson *et al.* [32], [27] use Bayes nets for hypothesis generation, their system incurs the overhead of tracking the region of interest through successive frames. Our earlier work [16], [17] uses a novel hierarchical knowledge representation scheme which not only ensures a low-order polynomial-time complexity of the hypothesis generation process, it also plays an important role in planning the next view.
5) *Nature of the next view planning strategy*
The system should, preferably be on-line and reactive – the past and present inputs should guide the planning mechanism at each stage. While schemes such as [29], [16], [17] are on-line, that of Gremban and Ikeuchi [7] is not. An off-line approach may not always be feasible, due to the combinatorial nature of the problem. An on-line scheme may result in significant reduction of the search space. An on-line scheme has the additional capability to react to unplanned situations, such as errors.
6) *Uncertainty handling capability of the hypothesis generation mechanism*
Approaches such as those of Goldberg and Mason [33], Gremban and Ikeuchi [7], and Liu and Tsai [31] are essentially deterministic. An uncertainty-handling mechanism makes the system more robust and resistant to errors compared to a deterministic one. Dickinson *et al.* [32], [27], Borotschnig *et al.* [29] and our earlier system [16], [17] use Bayesian methods to handle

uncertainty, while Hutchinson and Kak [26] use the Dempster-Shafer theory. In the work of Callari and Ferrie [28], the ambiguity in super ellipsoid-modeled objects is a function of the parameters estimated, on the basis of which the next move is determined. Schiele and Crowley [30] use a transinformation-based mechanism to propose the next move.

However, all the above systems do not consider the following issues:

1) To the best of our knowledge, no existing object recognition system handles the case when the complete object does not fit into the camera's field of view. This changes the domain of the problem completely – necessitating a completely new knowledge representation scheme, and a next view planning strategy.
2) Further, no existing system handles the case when the internal parameters of the camera are changed, either unintentionally, or on purpose.

We propose a novel on-line active 3-D recognition scheme using an uncalibrated camera. It uses a hierarchical part-based representation scheme in conjunction with a probabilistic framework for recognition and planning. Our active next view planning-based scheme is suited for situations where the camera sees only a portion of a large 3-D object. (This allows the system to operate very close to an object of interest, for example.) An important feature of our work is the use of Inner Camera Invariants [1], [2] – this allows the recognition system to work in spite of unintentional or purposive changes in the internal parameters of an uncalibrated camera. We assume that an object is represented by a set of identifiable parts. Our system uses simple geometrical features in conjunction with any other type of feature (geometrical, colour, photometric, etc.) for characterizing parts. In contrast to our approach, volumetric primitives used in [27] are associated with a high feature extraction cost, while appearance-based methods [34], [35] require the object of interest to be segmented out from the background. The system setup and viewing geometry is the most general – 6 degrees of freedom between the camera and the object, and it is based on a commonly used projective camera model. The paper [36] presents a preliminary version of our system, while detailed explanations may be found in [3].

The authors' earlier work on active 3-D object recognition [16], [17] looks at a different problem - the entire objects lies in the camera's field of view. These papers consider a 1-DOF uncalibrated camera, and propose an aspect graph-based knowledge representation scheme and a probabilistic active recognition strategy. This paper tackles a harder problem, where the object may not fit into the camera's field of view. Moreover, this approach is independent of intentional/accidental changes in camera internal parameters, unlike the previous approach. The authors propose a novel hierarchical part-based knowledge representation scheme, and a new probabilistic active recognition scheme for this problem. The rest of the paper is organized as follows. Section II describes our method of pose estimation using Inner Camera Invariants. We describe our hierarchical part-based knowledge representation scheme in Section III. Section IV describes our scheme of object recognition through next view planning. We present results of experiments with our system, in Section V.

II. 3-D Euclidean Pose Estimation using Inner Camera Invariants

We use *Inner Camera Invariants* to estimate the pose of parts present in a view of an object. The system uses this information to plan the next view, if the given view does not correspond to a unique pose of a particular object.

A commonly used projective camera model is [37]:

$$\lambda \mathbf{m} = \mathbf{P}\mathbf{M} = \mathbf{A}\left[\mathbf{R} \mid \mathbf{t}\right]\mathbf{M} \tag{1}$$

Here, $\mathbf{M} = (X, Y, Z, W)^T$ is a 3-D world point, and $\mathbf{m} = (x, y, 1)^T$ is the corresponding image point. $\mathbf{R}$ (3×3) and $\mathbf{t}$ (3×1) are the rotation and translation aligning the world coordinate system with the camera coordinate system (the external camera parameters), and $\mathbf{A}$ is the matrix of the internal parameters of the camera (the focal lengths in the x and y directions f_x and f_y, the skew parameter s, and the principal point (u_0, v_0)):

$$\mathbf{A} = \begin{bmatrix} f_x & s & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

The skew parameter s may often be considered to be negligible [37], [38]. Suppose we know three 3-D points, $\mathbf{M}_p = (X_p, Y_p, Z_p, 1)^T$ and their images $\mathbf{m}_p = (u_p, v_p, 1)^T$, $p \in \{i, j, k\}$. Eliminating the internals of the camera,

$$\begin{cases} J_{ijk} = \frac{u_i - u_j}{u_i - u_k} = \frac{\frac{\mathbf{r_1 M}_i}{\mathbf{r_3 M}_i} - \frac{\mathbf{r_1 M}_j}{\mathbf{r_3 M}_j}}{\frac{\mathbf{r_1 M}_i}{\mathbf{r_3 M}_i} - \frac{\mathbf{r_1 M}_k}{\mathbf{r_3 M}_k}} \\ K_{ijk} = \frac{v_i - v_j}{v_i - v_k} = \frac{\frac{\mathbf{r_2 M}_i}{\mathbf{r_3 M}_i} - \frac{\mathbf{r_2 M}_j}{\mathbf{r_3 M}_j}}{\frac{\mathbf{r_2 M}_i}{\mathbf{r_3 M}_i} - \frac{\mathbf{r_2 M}_k}{\mathbf{r_3 M}_k}} \end{cases}, \tag{3}$$

where J_{ijk} and K_{ijk} are *image measurements* that are functions of $[\mathbf{R} \mid \mathbf{t}]$ $(= [\mathbf{r_1}\ \mathbf{r_2}\ \mathbf{r_3}]^T)$ and $\mathbf{M}_p$ $(p \in \{i, j, k\})$, and are independent of camera internals.

$$\begin{cases} J_{ijk} = f_{ijk}(\mathbf{R}, \mathbf{t}, \mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_k) \\ K_{ijk} = g_{ijk}(\mathbf{R}, \mathbf{t}, \mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_k) \end{cases} \tag{4}$$

J_{ijk} and K_{ijk} are *Inner Camera Invariants* – image-computable invariants of the homography $\mathbf{A}$. We describe Inner Camera Invariants in detail, in earlier works [1], [2], [3]. We show that Inner Camera Invariants can be used for many diverse visions applications – without going through an often cumbersome process of camera calibration, or explicitly estimating camera internal parameters (self-calibration). Two prominent areas are 3-D Euclidean pose estimation from knowledge about landmarks, and 3-D Euclidean reconstruction from known ego-motions. Such techniques are important for autonomous robot navigation, for example. In [1], [2], [3], we additionally show uses of Inner Camera Invariants in related applications – interpolation of camera motion, interpolation of image measurements, and obtaining both the motion and structure in special cases.

We emphasize that Inner Camera Invariants are a new class of invariants, not to be confused with Projective Invariants [13], [6], etc. Our method relies directly on 3-D

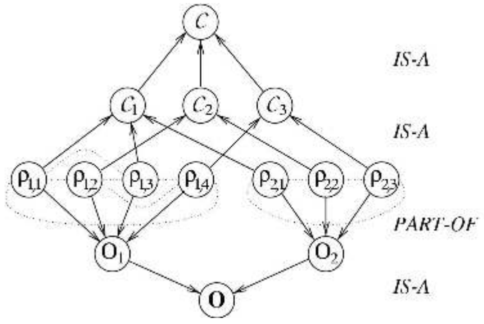

Fig. 3. The knowledge representation scheme: an example

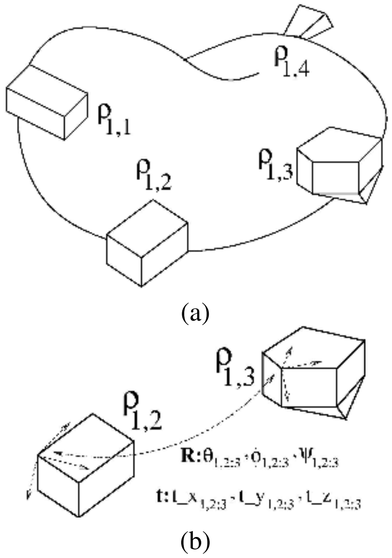

Fig. 4. (a) An example of an object O_1 with 4 parts. (b) A pair of parts is related by a rigid-body transformation $\mathbf{R}$ and $\mathbf{t}$ – three rotation and three translation parameters (details in text).

pose estimates (obtained through Inner Camera Invariants) to check consistency relations between a group of parts. In this paper, we use Inner Camera Invariants for estimating the pose of a part ($\mathbf{R}$ and $\mathbf{t}$). Suppose we know the Euclidean coordinates $(X_i, Y_i, Z_i, 1)^T$ of 5 points (in general position) in the world coordinate system. *Six* independent Inner Camera Invariant measurements give us six equations (of the type in (3)) in 6 unknowns: 3 rotations and translations each. We solve these equations to get the pose, using a suitable non-linear optimization routine (`constr/fmincon` in `MATLAB`). In [1], [2], [3], we also show two special cases where it is possible to obtain closed-form linear solutions for pose estimation. These, however impose a special structure on the landmarks used for pose estimation. For a 4-DOF system (*e.g.*, a setup with one rotational and all three translational DOF) as in Fig. 2, we adopt the same procedure with *four* independent (inner camera) invariant measurements from four equations. We discuss issues related to the robustness and stability of Inner Camera Invariants in a separate work [2]. For example, we show that to reduce the effect of pixel noise on the computation of J_{ijk} and K_{ijk}, the triplet of points must be appropriately chosen – the numerator and denominator should be of comparable order, and neither is too small. We also consider the effect of varying pixel noise (of the order of 1 and 2.5 pixels) on computations involving Inner Camera Invariants. We show that such pixel errors do not result in unbounded errors in the constraint equations for the optimization process.

III. The Knowledge Representation Scheme

Two major components of an active recognition scheme are – a planning algorithm (to plan the 'best' next view), and a suitable organization and representation of the objects in the model base (to facilitate planning and recognition). We propose a part-based hierarchical knowledge representation scheme that encodes domain knowledge about the objects in the model base. Fig. 3 illustrates an example of our knowledge representation scheme. We use the knowledge representation scheme for probability calculations, as well as planning the next view.

We consider a view of an object to contain 2-D or 3-D **parts** (which are detectable using 2-D or 3-D projective invariants, for example), and other 'blank' or 'featureless' regions (which the given set of feature detectors cannot identify). Thus, according to our formulation, an object is composed of parts, but is not partitioned into a collection of parts. Fig. 4(a) shows an example of an object O_1 (of Fig. 3), with parts $\rho_{1,1}$, $\rho_{1,2}$, $\rho_{1,3}$ and $\rho_{1,4}$. Let us consider vertices as the only features – O_1 is composed of the above 4 parts, and other 'featureless' regions.

- $\mathbf{O}$ represents the set of all objects $\{O_i\}$. An object node O_i stores its probability, $P(O_i)$
- An object O_i is composed of N_i parts. Thus, a part $\rho_{i,j}$ $(1 \leq j \leq N_i)$ has a *PART-OF* relationship with its parent object O_i. A part node stores the 3-D Euclidean structure of its n constituent vertices $(X_i, Y_i, Z_i)^T$, $1 \leq i \leq n$. (*e.g.*, $n \geq 5$ for a 6-DOF case and $n \geq 4$ for a 4-DOF case: Section II). Each part has a local coordinate system associated with it, with respect to which the coordinates are stored (Fig. 4(b)).
- A part node has $\mathbf{R}$ and $\mathbf{t}$ links with its nodes corresponding to its neighbouring parts. Fig. 4 (b) shows two of these parts, with the $\mathbf{R}$ and $\mathbf{t}$ relations between them – represented by the three rotation parameters $\theta_{1,2:3}$, $\phi_{1,2:3}$ and $\psi_{1,2:3}$, and the three translation parameters $t\_x_{1,2:3}$, $t\_y_{1,2:3}$ and $t\_z_{1,2:3}$.
- We define a **Part-Class** as a set of parts, equivalent with respect to a feature set. The set of parts is partitioned into different equivalence classes with respect to a given feature set: these equivalence classes are part-classes. $\mathcal{C}$ represents the set of all part-classes $\{\mathcal{C}_1, \mathcal{C}_2, \ldots \mathcal{C}_k\}$ for all parts belonging to the objects in the model base.
- We assume a function *PART_CLASS* to map the set of parts to the set of part-classes *i.e.*,

$$\mathit{PART\_CLASS} : \{\rho_{i,j}\} \longrightarrow \mathcal{C}$$

There is an *IS-A* relationship between a part, and its associated part-class. Thus, a part node $\rho_{i,j}$ has exactly one link with its corresponding part-class node $\mathcal{C}_k$, and the node for the object O_i, to which it belongs. In the

```
ALGORITHM identify_object_and_pose

              (* ------ FIRST PHASE ------ *)
1.  initialize_object_probs();  (* 1/N *)
2.  img:=get_image();
3.  part_class_info:=identify_part_classes(img);
    IF NO part_class observed THEN
          make random movement; GOTO step 2;
4.  search_tree_root:=
    const_search_tree_node(part_class_info,[$\mathbf{I}|\mathbf{0}$]);
5.  compute_hypothesis_probs(search_tree_root);
               (* Eq. 6 *)
6.  IF the prob of some hypothesis $\geq$ a
    pre-determined thresh THEN exit:success;
7.  expand_search_tree_node(search_tree_root,
      MAX_LEVELS); (* Sec IV-C *)
              (* ------ SECOND PHASE ------ *)
      previous:=search_tree_root;
      expected:=get_best_leaf(search_tree_root);
8.  {[$\mathbf{R}|\mathbf{t}$]}:=find_movement(expected,previous);
    make_movement({[$\mathbf{R}|\mathbf{t}$]});  img:=get_image();
9.  part_class_info:=identify_part_classes(img);
    IF NO part_class observed THEN
      undo_movement({[$\mathbf{R}|\mathbf{t}$]}); (* backtrack *)
      expected:=get_next_best_leaf(previous);
      GOTO step 8;
10. IF obs view $\not\equiv$ expected THEN
      new_node:=const_search_tree_node(
            part_class_info,{[$\mathbf{R}|\mathbf{t}$]}); ELSE
      modify_search_tree_node_with_observation(
            expected,part_class_info);
      new_node:=expected;
11. compute_hypothesis_probs(new_node);
12. IF the prob of some hypothesis $\geq$ a
    pre-determined thresh THEN exit:success;
13. expand_search_tree_node(new_node,MAX_LEVELS);
      expected:=get_best_leaf(previous);
      previous:=new_node;
14. GOTO step 8
```

Fig. 5. The Object Recognition and Pose Identification Algorithm

example of Fig.s 3 and 4, parts $\rho_{1,1}$ and $\rho_{1,3}$ belong to part-class $\mathcal{C}_1$.

IV. The Object Recognition Scheme

The system starts with an arbitrary view of an object in our model base. Our aim is to identify the given object, and the viewer pose with respect to it. There are two main components of our recognition scheme:

1) Hypothesis generation, and
2) Next view planning

Our scheme is independent of the particular technique to identify a part-class. The only requirement is that it should contain at least n points of interest for pose computation, $n = 5$ for the 6-DOF case, and $n = 4$ for the 4-DOF case (Section II). Fig. 5 describes the main steps in our algorithm. The first phase begins with initialization of all object probabilities. The system then takes an image of the given view, and identifies the part-classes corresponding to the parts present in the image. The next step is the formation of hypotheses about the identity of the observed parts. We describe our probabilistic hypothesis generation scheme in detail in Section IV-A. If the probability of some hypothesis is above a pre-determined threshold, then we exit and declare success. Otherwise, we invoke our search process to decide the best move from the current viewpoint, which will disambiguate between the competing hypotheses. Section IV-C describes the search process and the second phase of the object recognition algorithm, in detail.

A. Hypothesis Generation

Let the given view of an object contain m parts – ρ_{i,j_1}, ρ_{i,j_2}, $\ldots$ ρ_{i,j_m}. This view could correspond to any of the n objects in the model base. Further, this configuration of parts could have come from many different positions within the same object O_i. From the image information, we can only identify the *part-classes* $\mathcal{C}_{k_1}$, $\mathcal{C}_{k_2}$, $\ldots$ $\mathcal{C}_{k_m}$ (where $\mathcal{C}_{k_p}$ and $\mathcal{C}_{k_q}$ are not necessarily different) corresponding to each observed part, respectively (*PART_CLASS*$(\rho_{i,j_p}) = \mathcal{C}_{k_p}$). The part-classes may be identified by using 2-D or 3-D projective invariants, possibly in conjunction with some other non-geometric features such as grey level or colour information, reflectance ratio values, etc. One can also use to advantage Lowe's work on perceptual grouping *e.g.,* [15] – The basic aim is to derive groupings or structures in an image that are likely to be invariant over wide ranges of viewpoints. A more recent publication [14] describes an approach to indexing without using (projective) invariants – this can also be used to advantage in identifying a part-class. We emphasize however, that our scheme is independent of the particular technique to identify a part-class.

The system generates different part configuration hypotheses corresponding to the given view: We compute the estimated pose of each part (Section II), and check if the relative poses of each part in the configuration are consistent with the **R** and **t** values in the knowledge base, within error limits (for our experiments with the architectural models for example, we use ± 5° and ± $20mm$, respectively). *Thus, the part pose estimation phase itself helps in a first-level pruning of the list of competing view interpretation hypotheses.* This also offers a simple method to offset small inaccuracies in the part pose estimation process (Section II). *Thus, one does not need to use joint projective invariants between observed parts – our method relies directly on 3-D pose estimates to check consistency relations between a group of parts.* The next section describes the process of computing probabilities associated with each part configuration hypothesis.

B. a priori *Probability Calculations*

For N objects in the model base, the *a priori* probability of each object before taking the first observation, is $1/N$. We need estimates of the *a priori* probabilities of different configurations of parts that may occur (Step 1 in Fig. 5)

$$P(\rho_{i,j_1},\ \rho_{i,j_2},\ \ldots\ \rho_{i,j_m}) = P(O_i)\ \cdot\ P(\rho_{i,j_1},\ \rho_{i,j_2},\ \ldots\ \rho_{i,j_m} \mid O_i) \qquad (5)$$

We may form estimates of $P(\rho_{i,j_1},\ \rho_{i,j_2},\ \ldots\ \rho_{i,j_m} \mid O_i)$ from a very large number of views of the given object from different positions, and different values of the internals of the camera (the focal length, for example on which the field of view of the camera depends) — this is done *off-line*, before taking the first observation.

However, a satisfactory estimation of *a priori* probabilities using this method, may not be easy. If we make some assumptions about the nature of the 3-D object models, we can formulate an approximate method to estimate the *a priori* conditional part-configuration probabilities. Let us consider the domain of objects with planar parts. For such a case, one may approximate the *a priori* probability of a part configuration by its relative area in the 3-D model, not on any image-based features. The rationale behind this approximation is as follows. Ideally one would need a very large number of observations to get satisfactory *a priori* probability estimates. The camera pose would need to be sampled from the space of internal and external camera parameter values. For external parameters for example, one would have as many observations with the camera looking at the object from the right side, as would be from the left. Hence, one may have a good estimate of the *a priori* probability by looking at the part configuration head-on. The camera field of view depends on its focal length, an internal parameter. Intuitively, a larger part is more likely to be visible in a larger number of observations, as compared to a smaller one. Thus, one may consider the *a priori* probability of observing the part proportional to its area in the 3-D model.

We emphasize that the probabilistic analysis in this paper (Section IV-B) *does not depend* on the specific method used to compute *a priori* probabilities - any method would do. Our implementation of a prototype system uses the above approximation (Section V).

1) a posteriori *Probability Computations:* We use the Bayes rule to compute the *a posteriori* probability of each hypothesized configuration (Step 5 in Fig. 5)

$$\begin{aligned} P(\rho_{i,j_1},\ \rho_{i,j_2},\ \ldots\ \rho_{i,j_m} \mid \mathcal{C}_{k_1},\ \mathcal{C}_{k_2},\ \ldots\ \mathcal{C}_{k_m}) = \\ P(\rho_{i,j_1},\ \rho_{i,j_2},\ \ldots\ \rho_{i,j_m})\ \cdot \\ P(\mathcal{C}_{k_1},\ \mathcal{C}_{k_2},\ \ldots\ \mathcal{C}_{k_m} |\ \rho_{i,j_1},\ \rho_{i,j_2},\ \ldots\ \rho_{i,j_m})\ / \\ \sum [\ P(\rho_{l,j_1},\ \rho_{l,j_2},\ \ldots\ \rho_{l,j_m})\ \cdot \\ P(\mathcal{C}_{k_1},\ \mathcal{C}_{k_2},\ \ldots\ \mathcal{C}_{k_m} |\ \rho_{l,j_1},\ \rho_{l,j_2},\ \ldots\ \rho_{l,j_m})\ ] \end{aligned} \tag{6}$$

The summation above is for all objects O_l, and all possible configurations of parts within the object. Because of the *IS-A* relation between a part and a part-class in our knowledge representation scheme (Section III), each of the terms $P(\mathcal{C}_{k_1},\ \mathcal{C}_{k_2},\ \ldots\ \mathcal{C}_{k_m} |\ \rho_{l,j_1},\ \rho_{l,j_2},\ \ldots\ \rho_{l,j_m})$ is 1 for all parts belonging to a particular part-class and 0, otherwise.

We now compute the *a posteriori* probability of each object in the model base:

$$P(O_l) = \sum P(\rho_{l,j_1},\ \rho_{l,j_2},\ \ldots\ \rho_{l,j_m} |\ \mathcal{C}_{k_1},\ \mathcal{C}_{k_2},\ \ldots\ \mathcal{C}_{k_m}) \tag{7}$$

The summation is for all configurations of parts $\rho_{l,j_1}, \rho_{l,j_2}, \ldots\ \rho_{l,j_m}$ belonging to object O_l, which could have given rise to the given view containing part-classes $\mathcal{C}_{k_1}, \mathcal{C}_{k_2}, \ldots\ \mathcal{C}_{k_m}$. Each object node in the knowledge representation scheme uses Equation 7 to update its probability. In our hierarchical knowledge representation scheme, each part is linked to its neighbouring parts through $\mathbf{R}$ and $\mathbf{t}$ links. In the most general case, one may consider every part as the neighbour of every other part. For a given model base, we often observe that there are a large number of part pairs which cannot appear together in a given view (for example, parts which lie on opposite faces of an object). This reduces the computation time to only a small fraction of $\binom{N_i}{m}$ which would otherwise have been required. The corresponding search tree node stores the probability of each part configuration. Hence, Equation 7 needs constant space complexity.

C. Next View Planning

If the probability of no hypothesis (6) is above a predetermined threshold, we have to take the next view to try to disambiguate between the competing hypotheses. One needs to plan the best move out of the current state to disambiguate between the competing hypotheses, subject to memory and processing limitations, if any.

We describe the state of the recognition system in terms of the following parameters:

1) The competing view interpretation hypotheses, and
2) The set of $\mathbf{R}$ and $\mathbf{t}$ movements made thus far.

We use a search tree node to represent the system state. Search tree expansion proceeds according to the $\mathbf{R}$ and $\mathbf{t}$ relations in the knowledge representation scheme. Each search tree move is to get to the centre/centroid of the expected part. *Thus, the expected part is more likely to be in the camera's field of view even in the event of a zoom-in/zoom-out. Additionally, this provides robustness to small movement errors.* The planning process aims to get to a leaf node of the corresponding search tree – one corresponding to a unique part-configuration. One may also employ a limited memory search tree expansion (MAX_LEVELS in Fig. 5) Search tree node expansion is always finite because the number of parts in any object is finite. Further, there are no cycles in the search tree. No part is repeated along any path in the search tree. Thus, there can be no search tree expansion indefinitely oscillating between a set of parts.

We use three stages of filtering to get the best leaf node (Step 7 in Fig. 5) There is a search tree node corresponding to the current observation - we expand this node using the above strategy. Each leaf node corresponds to a unique part-class configuration. The first level of filtering considers the most probable view interpretation in the observed node's hypothesis list, and takes the consequent leaf nodes. The algorithm assigns a weight s^{level} to each search tree node, where s represents the number of hypothesized view, and $level$ is the search tree level (depth) the node lies on. Each leaf node has a path weight corresponding to the sum of all node weights along the path from the observed node. The second level of filtering considers those leaf nodes with the minimum path weight. We resolve remaining ties in favour of one of with the least total rotational movement. In what follows, we discuss various aspects of our recognition strategy in relation to existing methods.

1) The Search Process: A Discussion: Grimson [39] canonizes different philosophies behind object recognition. While our technique has some similarities with the Hypothesize-and-Test and Interpretation Tree paradigms, it differs on may counts. Our stepwise refinement method is much more general than the unary and binary constraints in [39]. Moreover, for the 3-D constraints in [39], the extraction of 3-D features from images is not an easy task. Our method relies on the least constrained of all geometric features - points, and also caters to other non-geometric features. Lastly, the basic premise behind a verification stage is to solve for a global interpretation, given only local interpretations. In our case, we have complete 3-D pose (through the Inner Camera Invariants-based method), and global consistency, with respect to all information that we have at any stage.

At any stage, the recognition algorithm has to determine whether it might be better to take a next view. A *decision theoretic agent* [40] selects the action with the highest expected utility – the Maximum Expected Utility (MEU) principle. The computations involved are often prohibitive [40]. Rimey and Brown [41] use decision theory for scene interpretation. They state that the next action to execute requires sequences of future actions to be considered, and that it is not feasible to enumerate all sequences of actions.

Our search process is consistent with a decision theoretic approach. We have a synergistic combination of reactive behaviour as well as planning. Intuitively, our utility is in getting a move with a maximum discriminatory ability – through the three levels of filtering to get to the best leaf node. We emphasize here that our algorithm finds the next action using a mechanism of look-ahead into possibilities, and not a sequence of actions. This is because our planning mechanism obtains the best distinguishing move at any particular stage, subject to memory and processing limitations. An interesting extension of our method may consider not just one best distinguishing move, but a set of such possibilities – whose 'cost' is below a particular threshold. Our strategy however, recovers from cases of incomplete planning because of its reactive nature – the re-planning after every observation. The decision on whether to take a new view or not depends on the probability of the interpretation hypotheses at any stage.

D. The Second Phase of the Object Recognition Algorithm

The previous section considers moves in the search space, in order to obtain the best distinguishing move. This section deals with the camera's actual movement (in accordance with the best distinguishing move).

The system makes the required movements $\{\langle R_x, R_y, R_z, t_x, t_y, t_z\rangle\}$, and takes an image at this position (Step 8). Similar to the process in Section IV-A, we generate different interpretation hypotheses corresponding to this view. The non-detection of some parts in the vicinity of the expected part (we do not predict a view) does not affect the system in any way. *This imparts robustness to the presence of clutter in an image.* If the current observation corresponds to the expected search tree node, we compute the probabilities of each view interpretation hypothesis. If the probability of some hypothesis is above a the predetermined threshold, we declare success, and exit (Step 12). If the current observation does not correspond to the expected search tree node, the system constructs a new node. This corresponds to the observed part-class information and the movement made thus far (Step 10). If no part-class is observed, we undo the current movements, get the next best leaf node, and proceed (Step 9). If the probability of no hypothesis is above the threshold, this node is expanded further (Step 14).

This illustrates the reactive nature of our strategy. The probabilistic hypothesis generation scheme (Section IV-A) incorporates all previous observations. If the observed view corresponds to the most probable view interpretation hypothesis at a particular stage, our search process uniquely identifies the object and its pose, in the following step (assuming no feature detection error for the expected part). Even if the observed view does not correspond to the most probable view interpretation hypothesis, our algorithm refines the list of hypotheses at each stage.

It is possible to compute an estimate of $T_{avg}(n)$: the average number of moves required for unique object and pose identification, given n competing part-configurations corresponding to the first view. Let us assume the entire space around the object to be divided into N_V viewing positions. Each of the N_V possible moves from the starting position partitions the part configuration hypothesis list, into equivalence classes. In general, any change in camera parameters (both external as well as internal) from a position could lead us to more than one part configuration. To serve as a benchmark, we can compute $T_{avg}(n)$ for a simple case of exactly one part-configuration being reachable from a point, and no errors in feature detection, or movement. We choose a move that partitions the initial set of part-configuration hypotheses into more than one equivalence class. (Section V-A lists a relevant case when such a move may not exist.) If the size of the part-configuration hypothesis list in one such equivalence class is j, the expected additional number of observations is $T_{avg}(j)$, where $1 \leq j < n$. Let us assume that j can take on any of the values 1 to $n-1$ with equal probability, We have $T_{avg}(n)$ = 1 + $\frac{\sum_{j=1}^{n-1} T_{avg}(j)}{n-1}$, and $T_{avg}(1)$ = 1. By induction, we can show that $T_{avg}(n)$ = $O(log_e n)$.

V. Experimental Results & Discussion

Our experimental setup has a camera system has 4 degrees of freedom - translations along the $\mathbf{X}$-, $\mathbf{Y}$- and $\mathbf{Z}$- axes, and rotation about the $\mathbf{Y}$- axis (as in Fig. 2). We have experimented with two model bases - architectural models (Fig. 1), and 8 buildings in the I.I.T. Bombay academic area. We have chosen as (2-D) parts the doors and windows of different shapes and sizes in the models. The first step in processing a given view of the object involves a segmentation of the image using sequential labeling [42]. Then we detect corners as intersection of lines on the boundaries of 'dark' regions. We use 2-D projective invariants using the canonical frame construction method [43] for recognizing all part-classes (except the 4-cornered ones – $DW4$ and $OPEN$ for which, we use the grey level information at a region near its centroid). We emphasize however, the *our recognition strategy is independent of the types of the parts and part-classes, or the method to detect them.* Model LH (Fig. 1(a)) has 167 parts, model DS (Fig. 1(b)) has 170, while model GH (Fig. 1(c)) has 122. Thus, *even though there are three models in our model base, we have chosen the models and the associated features such that there*

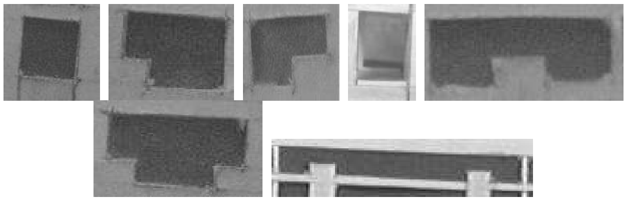

Fig. 6. The 7 part-classes which the 459 parts belong to, for our model base of architectural models: $DW4$, $DW6L$, $DW6R$, $OPEN$, $DW8HANDLE$, $DW8T$, and $DW12$, respectively in row-major order.

is a very high degree of interpretation ambiguity associated with a particular view of a few parts of the given object. Fig. 6 shows the 7 different part-classes these 459 parts (of different sizes) correspond to. The 7 part-classes, with the number of parts corresponding to each, are $DW4$(374), $DW6L$(24), $DW6R$(24), $OPEN$(21), $DW8HANDLE$(6), $DW8T$(6), and $DW12$(4), respectively. For our experiments with the I.I.T. Bombay buildings, we have chosen all windows of the type $DW4$ above. In this case, the uncertainty associated with a part-class is even larger - there are 1979 such parts in the 8 buildings considered.

We now briefly describe some representative experiments with the architectural models model base (Fig.s 7 – 11), and the I.I.T. Bombay buildings (Fig.s 12 – 14). For our experiments, adopt a strict criterion for program termination – we stop when there is exactly one hypothesis possible for the observed node. These experiments illustrate different features of our proposed recognition system namely, robustness to certain feature detection errors, the fact that parts could correspond to any 3-D configuration, invariance to zoom operations (invariance to internal camera parameter changes), and correct recognition even in the presence of clutter. We also discuss limitations of our proposed approach.

Experiments 1 and 2: Experiments 1 and 2 have a small degree of ambiguity corresponding to the first view. The initial view in Experiment 1 (Fig. 7), shows the two detected parts with part-classes $DW8T$ and $DW4$. The first view itself results the probabilities of the three models LH, DS and GH to be 1.000, 0.000 and 0.000, respectively. This is because the view could have come from only the first model. We do not stop here, however. Our algorithm will stop only when the probability of a particular part configuration hypothesis equals or exceeds a pre-determined threshold. For our experimentation, we have kept this at 1.000. This is the strictest possible limit, since the algorithm will stop only when there is exactly one part configuration hypothesis corresponding to the given view. Of the 6 possible hypotheses, our part pose estimation procedure (Section IV-A) prunes out 4 of them. The system plans a disambiguating move: the second view contains the expected part (bottom row, centre). This move results in correct recognition and pose estimation, in spite of *the failure to detect a neighbouring part* (top row, centre).

Experiment 2 (Fig. 8) shows another such example: the two windows on the left (corresponding to part-classes $DW8T$ and $DW4$, respectively) are not detected in the second planned view. In this case also, the first view uniquely determines the model present to be LH, with probability 1.000. The system stops only after the second view, which corresponds to exactly one part configuration hypothesis. The pose of the camera with respect to the identified part *LH_L_14* is $\langle\ -4.6^\circ, -2.54mm, 15.02mm, 139.98mm\ \rangle$.

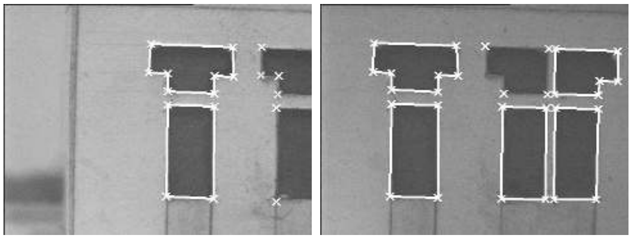

Fig. 7. Experiment 1: The sequence of moves required to identify the object and its pose. The failure to detect a part does not affect the system (details in text).

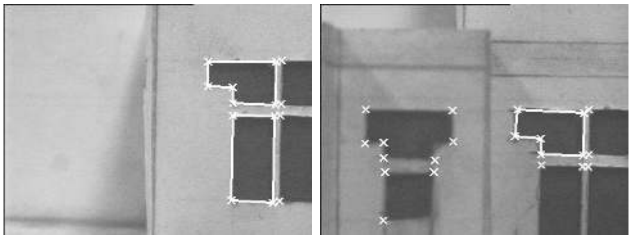

Fig. 8. Experiment 2: The sequence of moves required to identify the object and its pose. The failure to detect a part does not affect the system (details in text).

Experiment 3: For Experiment 3 and other succeeding experiments, the initial view has a high degree of ambiguity associated with it. The parts visible in a view need not come from the same plane. The initial view for Experiment 3 (Fig. 9) contains two parts lying on two faces at right angles to each other. There could be 374 hypotheses corresponding to a window corresponding to a part-class $DW4$. From the initial condition when each of the three objects could be present with equal probability, this state gets the probabilities of the three models LH, DS and GH as 0.162, 0.299 and 0.539, respectively. Part pose estimation results in a pruning of the hypothesis list corresponding to these two parts, to a hypothesis list of size 87. The system plans a move to disambiguate between the different hypotheses. This corresponding move takes us to a view (the second image in Fig. 9), whose view interpretation is unique. The probability of the three objects LH, DS and GH are 0.000, 0.000 and 1.000, respectively. This experiment illustrates that *the visible parts could have come from any 3-D configuration* – this does not affect the

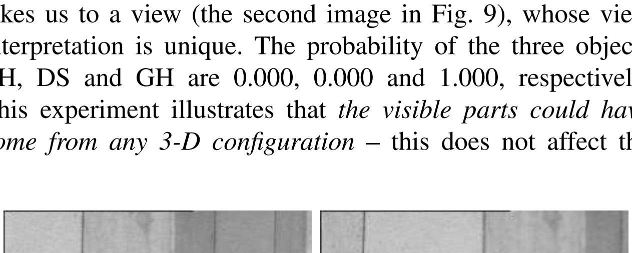

Fig. 9. Experiment 3: The sequence of moves required to identify the object and its pose. The parts in the initial view do not lie in the same plane.

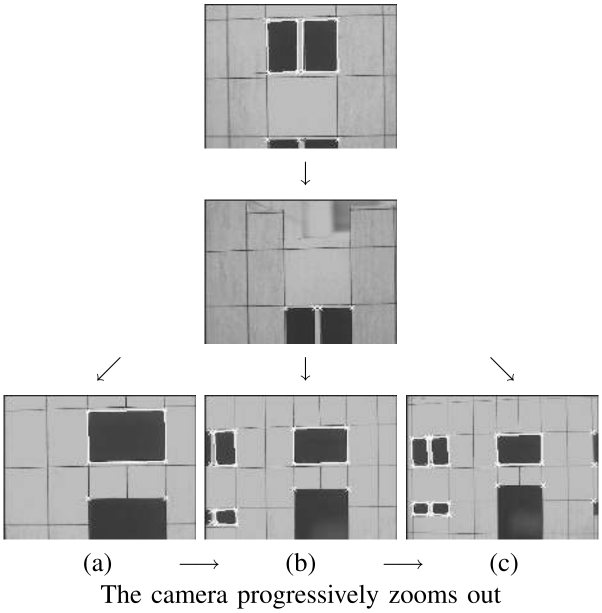

Fig. 10. Experiment 4: For the same first two views, we progressively zoom-out the camera in three stages. (a), (b) and (c) depict the three views which the camera sees, for the third view. This does not affect the recognition system in any way (details in text).

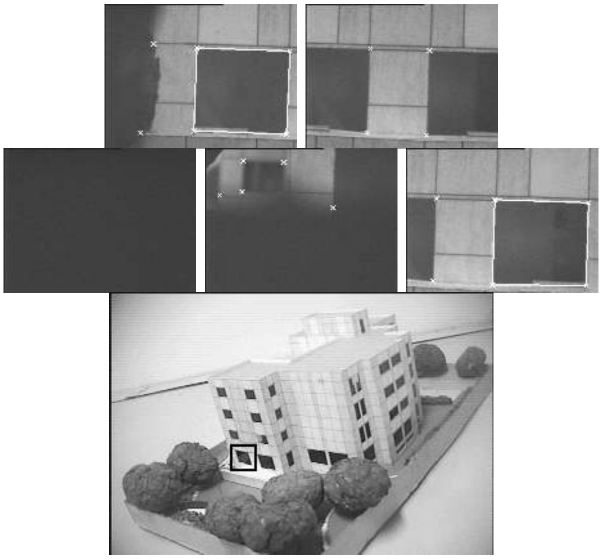

Fig. 11. Experiment 5: The sequence of moves (in row major order) required to identify the object and its pose. The first, third and fourth views are cluttered by the presence of a tree. The image at the bottom shows an overall view. The trees are in the foreground. The corresponding window is highlighted with a black square.

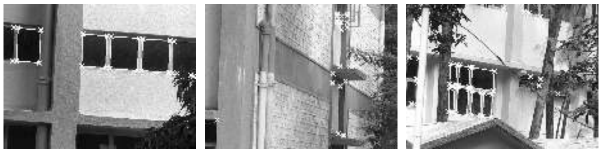

Fig. 12. Experiment 6 (I.I.T. Bombay Buildings): Backtracking on reaching a view without any part (details in text), and successful final recognition.

recognition system in any way.

Experiment 4: The use of Inner Camera Invariants for pose estimation allows us to *consider situations where the internal parameters of the camera may be varied on purpose, or unintentionally.* The first view in Fig. 10 could have come from 257 configurations of two adjacent parts with part-class $DW4$. Two moves from this position were sufficient to recognize the object, the third view containing the expected part (the large 4-cornered window, *GH_W_15*). For the same first two views, we performed two zoom-out operations at the the third camera position. The recognition results are the same in each of the cases — Fig. 10 (a), (b) and (c). Further, the camera pose with respect to part *GH_W_15* in these three cases are
$\langle\ 9.425^\circ,\ -22.000mm,\ -9.999mm,\ 150.000mm\ \rangle$,
$\langle\ 9.888^\circ,\ -22.000mm,\ -9.999mm,\ 150.000mm\ \rangle$, and
$\langle\ 9.896^\circ,\ -22.000mm,\ -9.999mm,\ 150.000mm\ \rangle$, respectively. Thus, accidental or purposive changes in internal camera parameters does not affect our system in any way.

It is interesting to compare this with Lowe's work on object recognition from local scale-invariant features [44]. Even though we have not considered any specific scale-invariant features, the same feature detector gives accurate results over a reasonably wide range of zoom-out operations.

We emphasize that the zoom-out operation were not performed in a graduated or pre-calculated manner – they were arbitrary. It is important to additionally note that the system did not plan these zoom-out operations – these were arbitrarily effected to test the system's resilience to variations (intentional / unintentional) in camera internal parameters. An interesting extension of this work would be to incorporate purposive changes in camera internal parameters as well – since any feature extraction routine has practical limits within which it works optimally. These purposive changes would be an important tool in the planning algorithm.

Experiment 5: In Experiment 5, the presence of a tree (an unmodeled object) accounts for clutter in the first, third and fourth view of Fig. 11. For Experiment 5, initially the three objects had a probability of 0.333 each. Following the first observation, the probabilities of LH, DS and GH were 0.172, 0.291 and 0.537, respectively. It is only with the final view that the probabilities change to 0.000, 0.000 and 1.000, respectively. The system plans the next move on the basis of a part: it does not predict an entire view. In these experiments, *recognition performance is not affected by the presence of unmodeled objects (clutter)* or the non-detection of parts in the vicinity of the expected part. 5 views are needed for unambiguous recognition and pose estimation. The size of the hypothesis list corresponding to the first view is 304.

Experiments 6 – 9: Buildings in I.I.T. Bombay: For Experiments 6 – 9, we have chosen an extremely difficult operating environment – there are numerous trees and other unmodeled objects. In addition to these, occlusions and lighting conditions also affect the performance of the system, as shown in the following experiments. In Experiment 6 (Fig. 12), the tree occludes the rightmost window, and the second and third windows from the left receive a wrong $\mathbf{X}-$ pose estimate due to occlusion from the pipe and the jutting wall, respectively. The planning on the basis of the available information from two windows leads to a region with no identifiable part (the

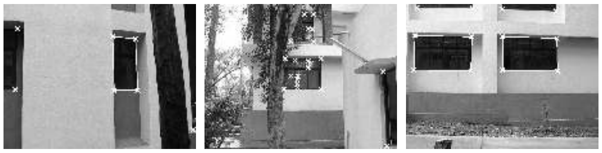

Fig. 13. Experiment 7 (I.I.T. Bombay buildings): Catastrophic failure - the effect of an occlusion (left), and reflection on the window panes and tree (centre) (details in text.)

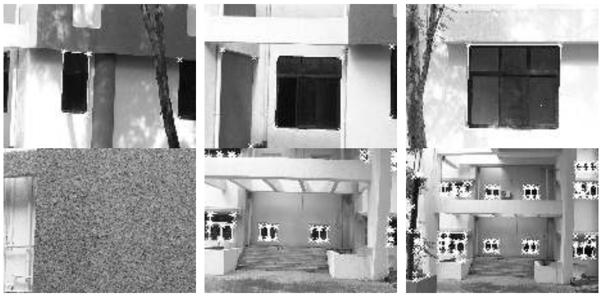

Fig. 14. Experiment 9: The sequence of moves (in row major order) required to identify the object and its pose (details in text.)

middle image). The system backtracks (Step 9 in the algorithm: Fig. 5), and takes the next move. This experiment shows the opportunistic nature of our system. The planning was with respect to the second most probable hypothesis, which does not correspond to the observed third view. However, the observed part-class configuration is unique for this set of moves, leading to successful recognition and pose estimation. Experiment 7 (Fig. 13) shows a case of catastrophic failure on two counts. We have chosen a difficult angle of imaging to start with – the jutting wall occludes part of the window, leading to a wrong $\mathbf{X}-$ pose estimate. The uncertainty list corresponding to the first view has all 1979 entries (since one part is observed). While the wrong pose estimate does not have an adverse effect on the second view owing to the camera being kept in a large field of view mode, the system fails to recognise any part (owing to reflections on the window panes, and the presence of the tree). The next planned view leads to the image on the right. For this model base, this sequence of moves cannot lead to this part configuration, and the system fails. For Experiment 9 (Fig. 14), the system needed 6 moves to recognise the part configuration and pose correctly. Starting from an uncertainty list of 1979 corresponding to the first view, the next two moves narrow it down to 32. On getting to a view without any part (the fourth figure), and a subsequent a backtrack, the uncertainty list has 2 entires for the fifth view. In this case, the system correctly identifies the object, but there were two competing part configurations. The final move of raising the camera up by $155cm$ resolves this ambiguity.

A. Limitations of the Proposed Approach

Our approach is not guaranteed to succeed for objects which have a similar layout of parts, with the corresponding parts in the objects corresponding to the same part-class. A primary limitation of our approach is the computation time involved: computation of Inner Camera Invariants (Section II) involves iterative nonlinear optimisation. A planning step (including the above computation time) takes up to about 13 minutes on a 700MHz PIII machine running Linux. While the system is quite robust to small movement errors, experiments with real buildings indicate that the system may also fail in the presence of a very large number of unmodeled objects, and failure to detect features. (The latter can cause *any* approach to fail).

VI. Conclusions

We present a new on-line scheme to identify large 3-D objects which do not fit into a camera's field of view (which allows the system to operate very close to the 3-D object), and finds the pose of the (uncalibrated) camera with respect to the object. The system does not assume any knowledge of the internal parameters of the camera, or their constancy (permitting a zoom-in/zoom-out operation, for example). The part-based knowledge representation scheme is used both for probabilistic hypothesis generation, as well as in planning the next view. We show results of successful recognition and pose estimation even in cases of a high degree of interpretation ambiguity associated with the initial view. The significance of using Inner Camera Invariants is the robustness of the system to internal camera parameters changes – accidental, or purposive. An interesting extension of this work is to use purposive internal camera parameter changes for planning the next view – one can zoom in to get further details, or zoom out, to get a wider field of view. The automatic learning of equivalent part classes in this context is another separate interesting extension.

VII. Acknowledgments

We are thankful to the reviewers for their constructive suggestions, which have greatly helped in improving the quality of this paper.

References

[1] M. Werman, S. Banerjee, S. Dutta Roy, and M. Qiu, "Robot Localization Using Uncalibrated Camera Invariants," in *Proc. IEEE International Conference on Computer Vision and Pattern Recogntion (CVPR)*, 1999, pp. II: 353 – 359.

[2] M. Werman, M. Qiu, S. Banerjee, and S. Dutta Roy, "Inner Camera Invariants and their Applications," Department of Computer Science and Engineering, I.I.T. Delhi, Tech. Rep., August 2001, http://www.cse.iitd.ac.in/∼suban/papers/inner07.pdf.

[3] S. Dutta Roy, "Active Object Recognition through Next View Planning," Ph.D. dissertation, Department of Computer Science and Engineering, Indian Institute of Technology, Delhi, 2000.

[4] P. J. Besl and R. C. Jain, "Three-Dimensional Object Recognition," *ACM Computing Surveys*, vol. 17, no. 1, pp. 76 – 145, March 1985.

[5] R. T. Chin and C. R. Dyer, "Model Based Recognition in Robot Vision," *ACM Computing Surveys*, vol. 18, no. 1, pp. 67 – 108, March 1986.

[6] A. Zisserman, D. Forsyth, J. Mundy, C. Rothwell, J. Liu, and N. Pillow, "3D Object Recognition using Invariance," *Artificial Intelligence*, vol. 78, pp. 239 – 288, 1995.

[7] K. D. Gremban and K. Ikeuchi, "Planning Multiple Observations for Object Recognition," *Int. Journal of Computer Vision*, vol. 12, no. 2/3, pp. 137 – 172, April 1994.

[8] B. V. Funt and G. D. Finlayson, "Color Constant Color Indexing," *IEEE Trans. on Pattern Anal. and Machine Intell.*, vol. 17, pp. 522 – 529, 1995.

[9] S. K. Nayar and R. M. Bolle, "Reflectance Based Object Recognition," *Int. Journal of Computer Vision*, vol. 17, no. 3, pp. 219 – 240, March 1996.

[10] S. J. Maybank, "Relation Between 3D Invariants and 2D Invariants," *Image and Vision Computing*, vol. 16, no. 1, pp. 13 – 20, 1998.

[11] R. Choudhury, S. Chaudhury, and J. B. Srivastava, "A Framework for Reconstruction based Recognition of Partially Occluded Repeated Objects," *Journal of Mathematical Imaging and Vision*, vol. 14, no. 1, pp. 5 – 20, February 2001.

[12] R. Choudhury, J. B. Srivastava, and S. Chaudhury, "Reconstruction-Based Recognition of Scenes with Translationally Repeated Quadrics," *IEEE Trans. on Pattern Anal. and Machine Intell.*, vol. 23, no. 6, pp. 617 – 632, June 2001.

[13] J. B. Burns, R. S. Weiss, and E. M. Riseman, "The Non-Existence of General-Case View-Invariants," in *Geometric Invariance in Computer Vision*, A. Zisserman and J. Mundy, Eds. MIT Press, 1992.

[14] J. S. Beis and D. G. Lowe, "Indexing without Invariants in 3D Object Recognition," *IEEE Trans. on Pattern Anal. and Machine Intell.*, vol. 21, no. 10, pp. 1000 – 1015, 1999.

[15] D. G. Lowe, *Perceptual Orgnization and Visual Recognition*. Kluwer Academic Publishers, 1985.

[16] S. Dutta Roy, S. Chaudhury, and S. Banerjee, "Isolated 3-D Object Recognition through Next View Planning," *IEEE Trans. on Systems, Man and Cybernetics - Part A: Systems and Humans*, vol. 30, no. 1, pp. 67 – 76, January 2000.

[17] ——, "Aspect Graph Based Modeling and Recognition with an Active Sensor: A Robust Approach," *Proc. Indian National Science Academy, Part A*, vol. 67, no. 2, pp. 187 – 206, March 2001, special Issue on Image Processing, Vision and Pattern Recognition.

[18] Y. Aloimonos, I. Weiss, and A. Bandopadhyay, "Active Vision," *Int. Journal of Computer Vision*, vol. 1, no. 4, pp. 333 – 356, 1987.

[19] R. Bajcsy, "Active Perception," *Proceedings of the IEEE*, vol. 76, no. 8, pp. 996 – 1005, August 1988.

[20] D. H. Ballard and C. M. Brown, "Principles of Animate Vision," *Computer Vision, Graphics and Image Processing: Image Understanding*, vol. 56, no. 1, pp. 3 – 21, July 1992.

[21] K. A. Tarabanis, P. K. Allen, and R. Y. Tsai, "A Survey of Sensor Planning in Computer Vision," *IEEE Trans. on Robotics and Automation*, vol. 11, no. 1, pp. 86 – 104, February 1995.

[22] S. Dutta Roy, S. Chaudhury, and S. Banerjee, "Active Recognition through Next View Planning: A Survey," *Pattern Recognition*, vol. 37, no. 3, pp. 429 – 446, March 2004.

[23] W. E. L. Grimson, "Sensing Strategies for Disambiguating Among Multiple Objects in Known Poses," *IEEE Journal of Robotics and Automation*, vol. RA-2, no. 4, pp. 196 – 213, 1986.

[24] C. B. Madsen and H. I. Christensen, "A Viewpoint Planning Strategy for Determining True Angles on Polyhedral Objects by Camera Alignment," *IEEE Trans. on Pattern Anal. and Machine Intell.*, vol. 19, no. 2, pp. 158 – 164, February 1997.

[25] J. Maver and R. Bajcsy, "Occlusions as a Guide for Planning the Next View," *IEEE Trans. on Pattern Anal. and Machine Intell.*, vol. 15, no. 5, pp. 76 – 145, May 1993.

[26] S. A. Hutchinson and A. C. Kak, "Planning Sensing Strategies in a Robot Work Cell with Multi-Sensor Capabilities," *IEEE Trans. on Robotics and Automation*, vol. 5, no. 6, pp. 765 – 783, December 1989.

[27] S. J. Dickinson, H. I. Christensen, J. Tsotsos, and G. Olofsson, "Active Object Recognition Integrating Attention and View Point Control," *Computer Vision and Image Understanding*, vol. 67, no. 3, pp. 239 – 260, September 1997.

[28] F. G. Callari and F. P. Ferrie, "Active Recognition: Using Uncertainty to Reduce Ambiguity," in *Proc. International Conference on Pattern Recognition (ICPR)*, 1996, pp. 925 – 929.

[29] H. Borotschnig, L. Paletta, M. Prantl, and A. Pinz, "Active Object Recognition in Parametric Eigenspace," in *Proc. British Machine Vision Conference (BMVC)*, 1998, pp. 629 – 638.

[30] B. Schiele and J. L. Crowley, "Transinformation for Active Object Recognition," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 1998, pp. 249 – 254.

[31] C.-H. Liu and W.-H. Tsai, "3D Curved Object Recognition from Multiple 2D Camera Views," *Computer Vision, Graphics and Image Processing*, vol. 50, pp. 177 – 187, 1990.

[32] S. J. Dickinson, H. I. Christensen, J. Tsotsos, and G. Olofsson, "Active Object Recognition Integrating Attention and View Point Control," in *Proc. European Conference on Computer Vision (ECCV)*, 1994, pp. 3 – 14.

[33] K. Goldberg and M. Mason, "Bayesian Grasping," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 1990, pp. 1264 – 1269.

[34] C. Y. Huang, O. I. Camps, and T. Kanungo, "Object Recognition Using Appearance-Based Parts and Relations," in *Proc. IEEE International Conference on Computer Vision and Pattern Recogntion (CVPR)*, 1997, pp. 877 – 883.

[35] O. I. Camps, C. Y. Huang, and T. Kanungo, "Hierarchical Organization of Appearance-Based Parts and Relations for Object Recognition," in *Proc. IEEE International Conference on Computer Vision and Pattern Recogntion (CVPR)*, 1998, pp. 685 – 691.

[36] S. Dutta Roy, S. Chaudhury, and S. Banerjee, "Recognizing Large 3-D Objects through Next View Planning using an Uncalibrated Camera," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2001, pp. II: 276 – 281.

[37] O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press, 1996.

[38] M. Pollefeys, R. Koch, and L. Van Gool, "Self-Calibration and Metric Reconstruction Inspite of Varying and Unknown Internal Camera Parameters," *Int. Journal of Computer Vision*, vol. 32, no. 1, pp. 7 – 25, August 1999.

[39] W. E. L. Grimson, *Object Recognition by Computer*. The MIT Press, 1990.

[40] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, Inc., 1995.

[41] R. D. Rimey and C. M. Brown, "Control of Selective Perception using Bayes Nets and Decision Theory," *Int. Journal of Computer Vision*, vol. 12, no. 2/3, pp. 173 – 207, April 1994, special Issue on Active Vision II.

[42] B. K. P. Horn, *Robot Vision*. The MIT Press and McGraw-Hill Book Company, 1986.

[43] C. A. Rothwell, "Recognition using Projective Invariance," Ph.D. dissertation, University of Oxford, 1993.

[44] D. G. Lowe, "Object Recognition from Local Scale-Invariant Features," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 1999, pp. 1150 – 1157.

Sumantra Dutta Roy completed his B. E. (Computer Engineering) from D.I.T. (currently N.S.I.T.), Delhi in 1993, and the M. Tech. and Ph. D. (Computer Science and Engineering) from I.I.T. Delhi in 1995 and 2001, respectively. He has been an Assistant Professor in the Department of Electrical Engineering at I.I.T. Bombay since 2001. He is a recepient of the Young Engineer Award of the Indian National Academy of Engineering for the year 2004. His research interests include Computer Vision and Image Processing, and Music Information Retrieval.

Santanu Chaudhury did his B. Tech. (1984) in Electronics and Electrical Communication Engineering and Ph. D. (1989) in Computer Science and Engineering from I.I.T. Kharagpur, India. Currently, he is a professor in the Department of Electrical Engineering at I.I.T. Delhi. He was awarded the Young Scientist Medal of the Indian National Science Academy in 1993. His research interests are in the areas of Computer Vision, Artificial Intelligence and Multimedia Systems.

Subhashis Banerjee did B. E. (Electrical Engineering) from Jadavpur University, Calcutta in 1982, and M. E. (Electrical Engineering) and Ph. D. from the Indian Institute of Science, Bangalore, in 1984 and 1989 respectively. Since 1989 he has been on the faculty of the Department of Computer Science and Engineering at I.I.T. Delhi where he is currently a Professor. His research interests include Computer Vision and Real-time Embedded Systems.