

Constraint-Fusion for Interpretation of Articulated Objects

Yacov Hel-Or

Department of Computer Science
The Weizmann Institute of Science
76100 Rehovot, Israel

Michael Werman

Department of Computer Science
The Hebrew University of Jerusalem
91904 Jerusalem, Israel

Abstract

This paper presents a method for interpretation of modeled objects that is general enough to cover articulated and other types of constrained models. The flexibility between components of the model are expressed as spatial constraints which are fused into the pose estimation during the interpretation process. The constraint fusion assists in obtaining the correct interpretation and in reducing the search of possible correspondences. The proposed method can handle any constraint (including inequalities) between any number of different components of the model. The framework is based on Kalman filtering.

1 Introduction

Estimating the pose of a 3D object from images or other sensed data is a classical problem in computer vision. Quite often, a model of the object is known and this information is used to estimate the pose of the object in the world. This problem is known as model-based pose determination and is used in many applications such as object recognition, object tracking, robot navigation, motion detection, etc. A complementary problem to the pose determination problem is the interpretation problem which deals with the correspondence between the given sensory data and the model features. This correspondence is necessary in localization procedures which are based on local features of the model. Both, the positioning and the interpretation problems are well documented in the literature (for reviews see [15, 16, 2]) however, the majority of the papers deal with 3D rigid objects and little attention has been given to articulated or constrained objects (e.g. [3, 6, 11, 13]).

An articulated object is an object composed of a set of rigid components connected at joints which allow certain degrees of freedom. These joints can be, for exam-

ple, *prismatic joints* which allow relative translation between components, or *revolute joints* which allow relative rotation of the components about a point. An example for such an object is a robot arm made up of several rigid components connected by movable joints. In this case, each model joint enforces a constraint on the spatial location of the body's components, thus, the problem of articulated objects is a special case of the general study of constrained models. We extend the definition of the problem to models that include additional general constraints such as co-linearity or co-planarity of the model components, angle relationships, etc, and also include inequality constraints such as a limited range of distances between points or a limited range of angles. We call these kind of models *constrained models*.

Existing methods that deal with constrained objects are restricted to deal with articulated models (e.g. [3, 6, 11, 13]). They deal with constraints that are due to prismatic or revolute joints. In this paper we present a general framework that can deal with all types of spatial constraints and is not limited to any particular type. Our method solves the interpretation and the localization problems simultaneously where constraints and measurements are considered and fused incrementally. In a previous paper [10] we described a method for the localization problem using incremental constraint fusion but there we assumed that the correspondence is given. Fusion of constraints into the pose determination of the components enables the information obtained on the pose of any single component to be propagated to all other components through the mutual constraints. In this manner, the estimated solution takes into account all the existing measurements and all the defined constraints. In this paper we focus on the interpretation problem and we explain the manner in which we integrate the correspondence search into the pose estimation process.

We deal here with models consisting of a set of feature points, such as maximum curvature, segment end-points or corners. The measurements taken on these points are noisy.

2 Formal Description of the Problem

A *constrained model* M of a 3D object consists of a set of rigid components $M = \{C_i\}_{i=1 \dots n}$. Each component C_i has its own local coordinate system and consists of a set of feature points whose locations are: $C_i = \{\mathbf{u}_{i,j}\}_{j=1 \dots m_i}$. The 3 dimensional vector $\mathbf{u}_{i,j}$ represents the location of the j^{th} point in the i^{th} component and is given in the local coordinate system of C_i . A set of points forming a component is rigid but the collection of components are not rigid. For each component C_i there is an associated parameter-vector \mathbf{T}_i representing the position of C_i relative to the viewer-centered frame of reference. Hence, \mathbf{T}_i is a six dimensional vector describing the location and the orientation of the local coordinates system of C_i relative to the viewer coordinates system. Since the components are restricted in their location due to flexible joints, the model includes, in addition to the representation of each component, a set of constraints which describe the mutual relationships between the components. These constraints are of the form: $\psi_k(\mathbf{T}_p, \mathbf{T}_q, \dots) = 0$. Each constraint may involve a single model component, such as a known location or a known orientation of the component, or several components as in the case of a revolute or prismatic joint between two components, a known distance between components, etc. Each constraint is expressed by an appropriate equation, for example, in an articulated constraint two components, C_p and C_q , are linked at a rotational point whose location is given by $\mathbf{u}_{p,i}$ in the local coordinates of C_p , and by $\mathbf{u}_{q,j}$ in the local coordinates of C_q . In such a case the constraint equation will be:

$$T_p(\mathbf{u}_{p,i}) - T_q(\mathbf{u}_{q,j}) = 0 \quad .$$

where T_i is the transformation function defined by the parameters in \mathbf{T}_i .

As previously mentioned, the model may also consist of inequality constraints of the form $\psi(\mathbf{T}_p, \mathbf{T}_q, \dots) > 0$. Let us assume, for the moment, that the constraints are restricted to equality constraints, and we will later describe the direct extension of these constraints to inequality constraints.

A *measurement* M' of a constrained object is rep-

resented by a collection of noise contaminated measurements and their uncertainties:

$$M' = \{(\hat{\mathbf{u}}'_{i,j}, \Lambda_{i,j})\}_{i=1 \dots n ; j=1 \dots m_i} \quad .$$

$\hat{\mathbf{u}}'_{i,j}$ - is a noise-contaminated measurement of the real location-vector $\mathbf{u}'_{i,j}$, associated with the j^{th} measured point of the i^{th} component. Both, $\hat{\mathbf{u}}'_{i,j}$ and $\mathbf{u}'_{i,j}$ are represented in a viewer-centered frame of reference. It is possible to have more than one measurement for a model point.

$\Lambda_{i,j}$ - is the covariance matrix depicting the uncertainty in the sensed vector $\hat{\mathbf{u}}'_{i,j}$. We do not constrain the dimensionality of the measured data but allow it to be 3D (stereo, range finder, etc.) or 2D (orthographic or perspective projection).

A *matching* (correspondence) between the model M and the measurement M' is a collection of pairs of the form

$$matching = \{\mathbf{u}_{i,j}, (\hat{\mathbf{u}}'_{i,j}, \Lambda_{i,j})\} \quad ,$$

which represents the correspondence between the model points and the measured points. For simplicity we mark every model point and its matched measurement with the same indices.

The problem :

Given a model M and a measurement M' , for each component C_i , find the measured points that correspond to its feature points and estimate its location \mathbf{T}_i . It is important to note that the solution $\{\mathbf{T}_i\}_{i=1 \dots n}$ *must* satisfy the model constraints:

$$\{\psi_k(\mathbf{T}_p, \mathbf{T}_q, \dots) = 0\}_{k=1 \dots r} \quad .$$

3 Background and Related Works

Extensive studies can be found in the literature dealing with pose estimation and interpretation of rigid objects from measurements, however, little attention has been given to articulated or constrained objects. Several studies can be found that deal with special cases of constrained objects, namely, articulated objects having prismatic or revolute joints, most of them in the context of recognition ([3, 6, 11, 13]). In general, the existing methods dealing with this problem can be divided into two main paradigms:

Divide and conquer methods:

The basic and naive method is to decompose the object into its parts and to estimate the pose of each component separately. In this method it is possible to follow the pose estimation by an assessment

of the current interpretation by testing whether the estimated position of neighboring components satisfy the constraints defined between them (up to a predefined threshold). Grimson [6, 5] follows this paradigm in order to identify a family of objects which differ in scale-factor, stretch factor or the angles between parts.

Although the simplicity of this method is attractive, it is unsatisfying from several aspects: Evaluating the pose of each component separately does not take into account the constraint information, therefore, each object component is located using only its measurements and no mutual information passes between parts. Additional information which can be obtained from measurements of neighboring components is not considered and thus, not all available information is exploited. Subsequently, the interpretation of each component is tested to be consistent with the interpretation of the neighboring parts (by verifying the satisfiability of the mutual constraints) but there is no way to verify whether the fulfillment of a current constraint does not deteriorate the fulfillment of the constraints of previously interpreted components.

Parametric methods:

It is possible to eliminate the defined constraints by decreasing the number of parameters that describe the pose of the object (so that the number of free parameters equals the degrees of freedom of the object). The remaining parameters are estimated during the estimation process, and the interpretation process is directed to be consistent with the current estimated parameters. Lowe [11] follows this method and estimates the free parameters of the viewpoint and of the model using Newton iterations. A similar method was used by Brooks [4] in the well known system *ACRONYM*. Mulligan et. al. [13] use the same approach for estimating the positions of an excavator’s arm. The main problem in the method of parameter reduction is the need for defining the dependence of each measurement on all the free parameters during the estimation process. The definition of the dependence is problematic for two reasons:

First, the complexity of this definition increases with the number of the body’s components. Second, in most cases, as the number of components of the object is greater, the order of the nonlinearity of the dependence equations is higher. This results in a more complex and less stable solution especially when using iterative methods based on linear approximation of the nonlinear equations (such as in [11]).

4 Constraints Fusion Method

In the two kinds of methods described in the last section there is no direct consideration of constraints in the pose estimation process; the constraints are not considered in the divide and conquer methods and they are eliminated, by reducing the number of estimated parameters, in the parametric methods. The method suggested in this paper considers both, measurements and constraints, in the estimation process. The pose of the object parts is estimated to conform optimally with the measurements while satisfying the model constraints. The measurements and the constraints are fused incrementally into the pose estimation enabling reliable and simple exploration of further interpretation. The method we suggest is a general scheme which overcomes the drawbacks of the other methods.

The idea is to treat both measurements and constraints similarly while varying only their associated uncertainty. The constraints are considered as perfect “measurements” with zero uncertainty whereas the measurements themselves (the actual measurements) have uncertainty greater than zero. In other words the actual measurements are considered *soft* constraints whereas the constraints are considered *strong*. The fusion of the actual measurements and the constraints during the pose estimation process is performed using the Kalman filter and it is in accord with [1]. The fusion of a constraint into the pose estimation enables us to predict the possible locations of associated components which have not been interpreted yet. Thus, simple comparison between this prediction and the possible measurements to be matched can eliminate irrelevant correspondences and helps us in the interpretation exploration. In order to simplify the explanation of the process we first elaborate the solution in the case where each of the object’s component consists of a single model point, and then we expand the solution to include multiple-point components.

5 Constrained Objects Having One Point Per Component

The simplest case of a constrained object is where each of its components consists of a single model point. In this case the object model is represented by: $M = \{C_k\}_{k=1 \dots n}$, where each component C_k has a single model point whose location is \mathbf{u}_k . Without loss of generality, we choose this point to be located at the origin of the local coordinates associated with C_k , i.e: $\mathbf{u}_k = (0, 0, 0)^t$. Measurements of the locations

of the model points are obtained. For simplicity assume n measurements are obtained, $\{(\hat{\mathbf{u}}'_i, \Lambda_i)\}_{i=1\dots n}$, a single measurement for each model point, represented in the viewer-centered coordinates. Additionally, assume in this case that the measurements are 3D data. The latter assumption is due to the inability to induce the 3D position of an isolated point from a single 2D measurement. The transformation of the k^{th} component, \mathbf{T}_k , is composed only of the translation vector $\mathbf{t}_k = (t_x, t_y, t_z)^t$ since the rotation part is irrelevant for an isolated point. Therefore, the general position vector, \mathbf{T} , to be estimated in such a case consists of the translation vectors of all the model components: $\mathbf{T} = (\mathbf{t}_1^t, \mathbf{t}_2^t, \dots, \mathbf{t}_n^t)^t$. Since the model points are located at the origin of the local coordinates the translation vector $\mathbf{t}_k = (x'_k, y'_k, z'_k)^t$ also describes the position of the k^{th} point in the viewer centered frame of reference. However, the evaluated estimation must satisfy a set of constraints: $\{\psi_j(\mathbf{T}) = 0\}_{j=1\dots r}$. For the specific case of an articulated object the constraints are:

$$\psi_j(\mathbf{t}_k, \mathbf{t}_l) = \|\mathbf{t}_l - \mathbf{t}_k\|^2 - d_{(k,l)}^2 = 0$$

where $d_{(k,l)}$ represents the constant Euclidean distance between two adjacent points, \mathbf{u}_k and \mathbf{u}_l , in the object.

5.1 Pose Estimation using An Incremental Process of K.F.

As stated, enforcing the model constraints into the pose solution is performed by considering the constraints as additional artificial “measurements” having zero uncertainty. The zero uncertainty of these “measurements” assures that the constraints are satisfied in the final solution. The estimation process is composed of an incremental refinement, for which at each step $k - 1$, there exists an estimate $\hat{\mathbf{T}}^{k-1}$ of the transformation \mathbf{T} and a covariance matrix Σ^{k-1} which represents the “quality” of the estimate $\hat{\mathbf{T}}^{k-1}$: $\Sigma^{k-1} = E\{(\hat{\mathbf{T}}^{k-1} - \mathbf{T})(\hat{\mathbf{T}}^{k-1} - \mathbf{T})^t\}$. Given a new measurement (which is a real one $(\hat{\mathbf{u}}'_k, \Lambda_k)$ or a constraint $\psi_k(\mathbf{T}) = 0$) the current estimate is updated to be $\hat{\mathbf{T}}^k$ with an associated uncertainty Σ^k . The accuracy of the estimate increases, as additional measurements are fused, i.e. $\Sigma^k \leq \Sigma^{k-1}$ ($\Sigma^{k-1} - \Sigma^k$ is nonnegative definite). Fusion of a constraint or a real measurement into the solution is performed, using the *extended Kalman filter* (E.K.F.) [12]. In addition *local iterations* [12] are performed in order to reduce the influence of the linearization effect on the final solution. Detailed explanation with full equations about the measurements and constraints fusion can be found in [8].

The sequential fusion of the measurements is possible due to the assumption that there is no correlation between the noise of different measurements (i.e. $\text{cov}\{\mathbf{u}_i, \mathbf{u}_j\} = 0$ where $i \neq j$). The incremental fusion of measurements gives us the ability to easily incorporate a matching (interpretation) process into the estimation process as will be described in the following section.

In some cases inequality constraints such as $g(x) \geq 0$ can be appear. Examples of such inequality constraints can be found in articulated models such as scissors and robot arms that are limited in the range of feasible angles between parts. In such a case we reduce the inequality constraint to equality by adding a slack variable. I.e. the inequality $g(x) \geq 0$, is rewritten as $g(x) - \lambda^2 = 0$, where λ is a new variable that is added to the state vector and is estimated during the filtering process.

Examples and results of applying the pose determination process on real and simulated data can be seen in [10].

6 The Measurement Interpretation

Using the incremental approach described above we adopt the techniques which solve the interpretation problem by a pruning search in the correspondence space. These techniques regard the correspondence problem as a search problem in a graph (Interpretation Tree). This graph defines a pairing between the model features and the measured features. The basic scheme behind these methods is to prune parts of the graph which represent impossible pairings.

Suppose we want to match the measurement $(\hat{\mathbf{u}}'_i, \Lambda_i)$ with the j^{th} model point. From the current estimate $(\mathbf{T}^{cur}, \Sigma^{cur})$, we extract an estimate of the location \mathbf{u}_j : $(\hat{\mathbf{t}}_j^{cur}, \Sigma_j^{cur})$ and evaluate the Mahanalo-bis distance between $\hat{\mathbf{t}}_j^{cur}$ and $\hat{\mathbf{u}}'_i$:

$$\delta = (\hat{\mathbf{t}}_j^{cur} - \hat{\mathbf{u}}'_i)(\Sigma_j^{cur} + \Lambda_i)^{-1}(\hat{\mathbf{t}}_j^{cur} - \hat{\mathbf{u}}'_i)^t$$

If δ is greater than a predefined threshold, the match is rejected. The greater the number of measurements and constraints fused prior to the match, the more precise is the estimate $\hat{\mathbf{t}}_j^{cur}$ and the elimination of irrelevant measurements is more effective. Therefore, there is great importance, in this method, to the order of the points being fused (matched) since before matching the j^{th} model point, we would like the system to obtain as much information as possible on the

location estimate $\hat{\mathbf{t}}_j$ so that the match verification is significant. Thus, at each step of the process the next point to be matched should be one associated with previously matched points through constraints, so that previous information (measurements and constraints) can be exploited. The following algorithm follows this idea.

Denote by ψ_k ($k = 1 \dots m$) the constraints of the model and by $point(\psi_k)$ the set of points on which ψ_k depends. The order of fusion of the measurements and the constraints is obtained from the following algorithm:

1. $FusedPoints = \emptyset$; $FusedConst = \emptyset$
2. $PointList = \{\hat{\mathbf{u}}'_1, \dots, \hat{\mathbf{u}}'_m\}$; $ConstList = \{\psi_1, \dots, \psi_r\}$
3. **while** ($ConstList \neq \emptyset$ **and** $PointList \neq \emptyset$) **do**
 - (a) **for each** ($\psi_k \in ConstList$ s.t. $point(\psi_k) \in FusedPoints$) **do**
 - i. fuse ψ_k
 - ii. delete ψ_k from $ConstList$ and add it to $FusedConst$
 - (b) **for each** ($\hat{\mathbf{u}}'_k \in PointList$ s.t. $\mathbf{u}_k \in point(FusedConst)$) **do**
 - i. match and fuse $\hat{\mathbf{u}}'_k$
 - ii. delete $\hat{\mathbf{u}}'_k$ from $PointList$ and add it to $FusedPoints$
 - (c) **if** there exists $\psi_k \in ConstList$ s.t. $(point(\psi_k) \cap FusedPoints) \neq \emptyset$ **do**
 - i. fuse ψ_k
 - ii. delete ψ_k from $ConstList$ and add it to $FusedConst$
 - (d) **else** select an arbitrary $\psi_k \in ConstList$ and **do** steps i-ii in (c).

In the case where a good match for the model point \mathbf{u}_j can not be found due to occlusion or inability to obtain information about certain interest points in the image, we synthesize an artificial measurement for the model point and associate it with an infinite uncertainty so that its influence on the rest of the process will be minimal. This scheme can also be helpful when we want to fuse a constraint ψ_k where some of its associated points $point(\psi_k)$ are unavailable.

7 Constrained Objects Having Multiple-Point Components

We easily extend the solution for objects having one point per component to objects that have multiple-point components. For every component C_k , one must estimate the transformation \mathbf{T}_k which is composed of a rotation part R_k and a translation part \mathbf{t}_k .

The process of evaluating all the transformations $\{\mathbf{T}_i\}$ is similar to the methods previously described for models having a single point per component, however the constraints are now associated with components rather than with single points. The information obtained from a measurement $\hat{\mathbf{u}}_{k,j}$ is fused into the solution \mathbf{T} with the same manner as we fuse measurements for a rigid object pose estimation. Detailed explanation of such a process can be found in [9]. The information obtained from a constraint is fused as described in Section 5.1. The order in which the measurements and constraints are fused follows the algorithm given in Section 6 with the following two changes:

The constraints are now associated with components rather than with single points, therefore, in the algorithm, the components $\{C_k\}$ replace the model points. Additionally, following the fusion of the constraints associated with component C_k , we find and fuse matched measurements for all the feature points $\{\mathbf{u}_{k,i}\}_{i=1 \dots m_k}$ (the feature points belonging to component C_k). The interpretation method for each component is similar to that for a rigid body, as presented in [9], where in our case there is additional *a priori* information about \mathbf{T}_k from the previously fused constraints. This order of interpretation ensures that prior to fusion of a point in any component, all available information from neighboring components and mutual constraints have been exploited in order to assist in rejecting irrelevant matches.

In the case where every component contains several model points, there is no need to restrict the measurements to be 3D since the pose of the component can be estimated from projections (2D measurements) [9].

8 Results

We applied our method to estimate the position of a real articulated 3D object from 2D images. The articulated model used, is a desk lamp shown in Figure 2, having 5 degrees of freedom. We consider the lamp model as a 23-point model (as shown in Figure 1) and we included the following constraints into the model:

- constant distance constraints between couples of points in the model (for example points 10 and 12).

- parallel constraints between 2 pairs of points (between points 10 and 12 and points 9 and 13).
- co-planar constraints between 4 or more points (points 10,12,13 and 9 are constrained to be coplanar).

Measurements of the 3D location of the points and the measurement uncertainty were obtained from stereo image pairs. This data is noisy due to digitization, inconsistent lighting and imprecise feature matching. The uncertainty due to noise were modeled according to the auto-correlation of the image features [14]. We estimated the pose of the lamp components from the noisy 3D measurements and from the constraints using our technique. The evaluated vector is a $23 \cdot 3$ dimensional location vector composed of the 23 locations of the model points. Figures 2a and 2b show 2 examples of lamp images having 2 different positions. Figures 2c and 2d show the corresponding results as synthetic images created from the estimated location vector. As can be seen, there is high correlation between the real model location and the synthesized reconstruction. More results on the pose estimation process can be seen in [10, 8].

The correspondence between the measured data and the model points was found according to the interpretation process we described. Figure 3 shows a limited part of the interpretation tree (I.T.) which is constructed for the desk lamp interpretation. This I.T. is used for the matching process as described in Section 6. Each node on the k th level of this I.T. represents a possible matching between the k th model point, as numbered in Figure 1, and some particular measured point. The measurements are numbered according to their real correspondence (i.e. the true match of the k th measured point is the k th model point). The score of each match is shown at the appropriate node where the value is the Mahalanobis distance δ calculated using the formula given in Section 6. For each level in the I.T. we show the three best scored nodes. The model constraints are fused during the parsing of the I.T. as described by the algorithm in Section 6. The distance constraints between model points k and j (denoted by $dist(k, j)$) are shown in the figure at the level at which they are fused. As can be seen the score of the correct matches are significantly lower than the erroneous matches. In the case where several measurements may approximately satisfy the fused constraints (such as points 6 and 10 which may both satisfy the constraint $dist(5, 6)$) the scores of these measurements will be small (see the scores of the measurements 6 and 10 in the 6th level) and the I.T. tree should be further explored in these directions.

9 Conclusion

This paper presented a framework based on Kalman filtering for model based pose estimation and interpretation that is general enough to cover articulated and other types of non-rigidly constrained models. The existing methods of pose estimation of constrained models, deal with articulated objects and with constraints that are due to prismatic or revolute joints between the model components. In our method we are not limited to any type of constraints and can deal with all types of constraints including inequalities. The proposed scheme enables an efficient simultaneous matching procedure which allows incremental fusion of additional matches that improve the pose estimation and the search complexity in the I.T.

There are computational aspects that were not covered in this paper such as methods to stabilize the convergence, parallelization techniques and methods to speed up the computation and to reduce the time complexity using Optimal Smoothing. This techniques is described in [7].

References

- [1] N. Ayache. *Artificial Vision for Mobile Robots Stereo Vision and Multisensory Perception*. MIT Press, 1991.
- [2] R. Basri. *The Recognition of 3-D Solid Objects from 2-D Images*. PhD thesis, Weizmann Institute of Science, Oct. 1990.
- [3] A. Beinglass and H.J. Wolfson. Articulated object recognition, or: How to generalize the generalized hough transform. In *CVPR*, pages 461–466, 1991.
- [4] R.A. Brooks. Model-based 3-D interpretation of 2-D images. *T-PAMI*, 5:140–150, 1983.
- [5] W.E.L. Grimson. Recognition of object families using parameterized models. In *ICCV*, pages 93–101, 1987.
- [6] W.E.L. Grimson. On the recognition of parametrized 2-D objects. *IJCV*, 2:353–372, 1989.
- [7] Y. Hel-Or. *Pose Estimation from Uncertain Sensory Data*. PhD thesis, Inst. of Computer Science, The Hebrew University of Jerusalem, 1993.
- [8] Y. Hel-Or and M. Werman. Recognition and localization of articulated and constrained objects. *submitted to the IJCV, October 1993*.

- [9] Y. Hel-Or and M. Werman. Absolute orientation from uncertain data: A unified approach. In *CVPR*, pages 77–82, 1992.
- [10] Y. Hel-Or and M. Werman. Model based pose estimation of articulated and constrained objects. In *the 3rd ECCV*, May, 1994.
- [11] D.G. Lowe. Fitting parametrized 3-D models to images. *T-PAMI*, 13:441–450, May 1991.
- [12] P.S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 1. Academic Press, 1979.
- [13] I.J. Mulligan, A.K. Mackworth, and P.D. Lawrence. A model-based vision system for manipulator position sensing. In *Workshop on Interpretation of 3D Scenes, Austin, Texas*, pages 186–193, 1989.
- [14] A. Shmuel and M. Werman. Active vision: 3D depth from an image sequence. In *ICPR*, pages 48–54, 1990.
- [15] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344, Aug 1987.
- [16] J.S.C. Yuan. A general photogrammetric method for determining object position and orientation. *IEEE Transactions on Robotics and Automation*, 5(2):129–142, 1989.

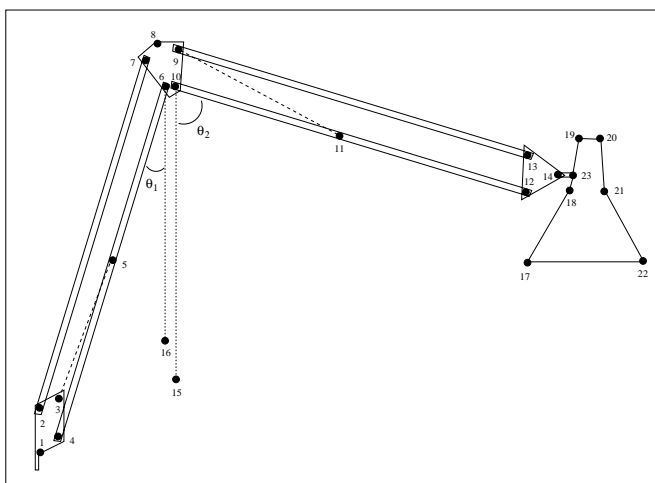


Figure 1: A schematic diagram of a lamp model having 23 points.

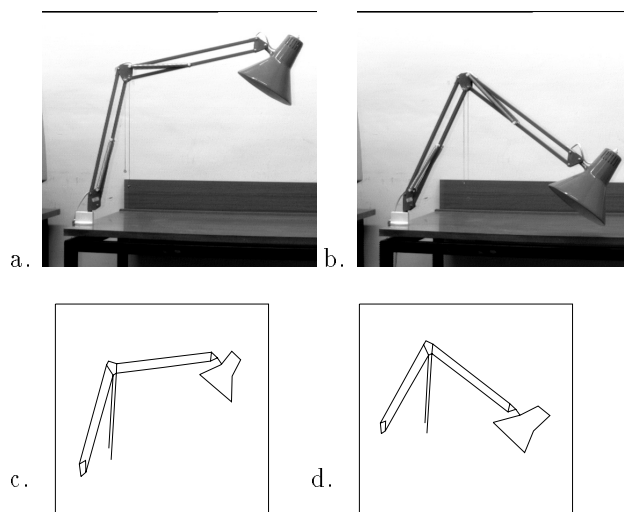


Figure 2: a-b) Images of a desk lamp at different positions. c-d) The corresponding result shown as a synthetic image created from the estimated location vector.

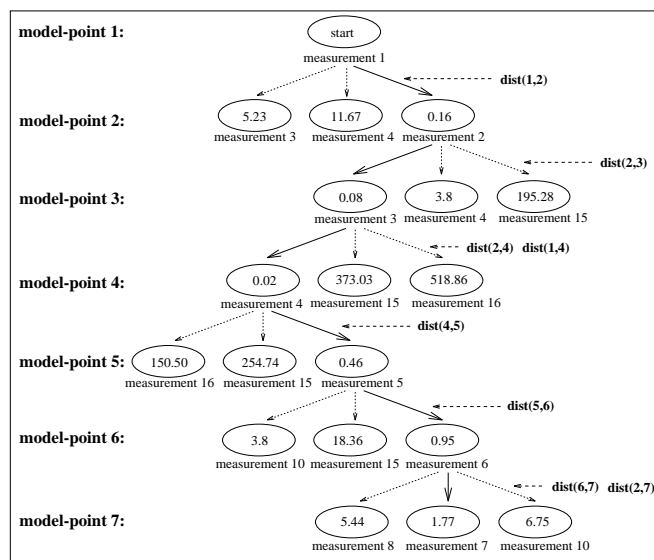


Figure 3: Results of the matching algorithm for the lamp model. A section of the pruned interpretation tree is displayed. Every level of the tree corresponds to one model point, and each node at a particular level corresponds to a possible match between the model point and a measured point. The score of each match is shown at the node where the value is the Mahalanobis distance of this match. For each level in the interpretation tree the three best scored nodes are shown. The distance constraints (denoted by $dist(k, j)$) are shown at the level at which they are fused.