Recognition and Localization of Articulated Objects

Yacov Hel-Or
Department of Applied Math.
The Weizmann Institute of Science
76100 Rehovot, Israel

Michael Werman
Institute for Computer Science
The Hebrew University of Jerusalem
91904 Jerusalem, Israel

Abstract

This paper presents a method for localization and 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 a precise and stable pose of each object's component and in finding the correct interpretation. 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]) 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, 5, 10, 12]).

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 example, *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, 5, 10, 12]). 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 which are not limited to any particular type. The method solves the interpretation and the localization problems simultaneously where constraints and measurements are considered and fused incrementally. 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 addition, the process enables a simple and efficient interpretation strategy. The fusion of the constraints and the measurements is performed using the Kalman filter.

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 $M = \{C_i\}_{i=1\cdots n}$. Each component C_i has its own local coordinate system and consists of a set of feature points whose locations are: $C_i = \{\mathbf{u}_{i,j}\}_{j=1\cdots m_i}$. 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: $\psi_k(\mathbf{T}_p, \mathbf{T}_q, \cdots) = 0$. 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: $T_p(\mathbf{u}_{p,i}) - T_q(\mathbf{u}_{q,j}) = 0$, 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 $\psi(\mathbf{T}_p,\mathbf{T}_q,\ldots) > 0$. 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 represented by a collection of noise contaminated measurements and their uncertainties:

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

$\hat{\mathbf{u}}'_{i,j}$ - 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 set 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 $\{\mathbf{T}_i\}_{i=1\cdots n}$ *must* satisfy the model constraints:

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

3 Background and Related Works

Extensive studies can be found in the literature dealing with pose estimation of rigid objects from measurements, however, little attention has been given to pose estimation of 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, 5, 10, 12]). 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 part separately. Grimson [5] and Shakunaga [13] follow this paradigm. In [5] the pose estimation of each part separately was followed by an assessment of the current interpretation of the parts by testing whether the parts satisfy the constraints defined between them (up to a predefined threshold). Although the simplicity of this method is attractive, it is obvious that it is unsatisfying since it does not exploit the fact that different components do belong to the same object, thus, no mutual information passes between parts and each part is located using only its measurements.

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.

For example, the pose of an articulated object in 2D having two components connected by a revolute joint, can be described by the translation and the orientation of each component (6 parameters) with an additional constraint due to the joint between the components

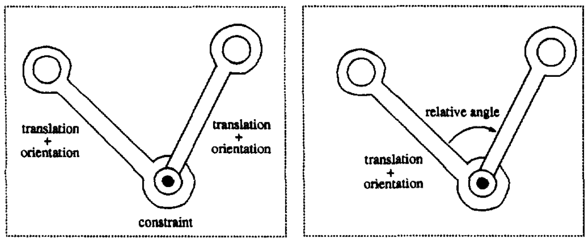

Figure 1: Two possibilities of parameter sets for the pose definition of an articulated object having a revolute joint.

(see Fig. 1). Alternatively, the pose of the object can be described by the translation and orientation of one of the components and the relative angle to the second component (4 parameters). The latter description eliminates the need to consider a constraint in the estimation process. Lowe [10] 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. [12] 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 [10]).

4 Constraints Fusion Method

In the two kind of methods described in the last section there are no direct consideration of constraints in the estimation process. Either, the constraints are not considered in the divide and conquer methods or 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 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]. In order to simplify the explanation of the pose determination of general constrained objects, 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: $M = \{C_k\}_{k=1\cdots n}$, 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, $\{(\hat{\mathbf{u}}_i', \Lambda_i)\}_{i=1\cdots n}$, 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: $\mathbf{T} = (\mathbf{t}_1^t, \mathbf{t}_2^t, \cdots, \mathbf{t}_n^t)^t$. 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: $\{\psi_j(\mathbf{T}) = 0\}_{j=1\cdots r}$. 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 Solving the System 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.) [11] which is a generalization of the Kalman filter to non-linear systems. In addition *local iterations* [11] 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. $\text{cov}\{\mathbf{u}_i, \mathbf{u}_j\} = 0$ if $i \neq j$). The incremental fusion of measurements rather than batch fusion (fusing all the measurements at once) is important since it 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. The initial *a priori* uncertainty associated with the parameter λ is infinite.

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 Mahanalobis 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. Before trying to match a measurement to a certain point we fuse as many constraints as possible associated with this point (and with previously matched points). Moreover, we choose the next point to be matched as that point having the largest number of constraints associated with previously matched points. A detailed algorithm for this matching process can be found in [6]. 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 are unavailable.

7 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 $\hat{\mathbf{T}}_k$ 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

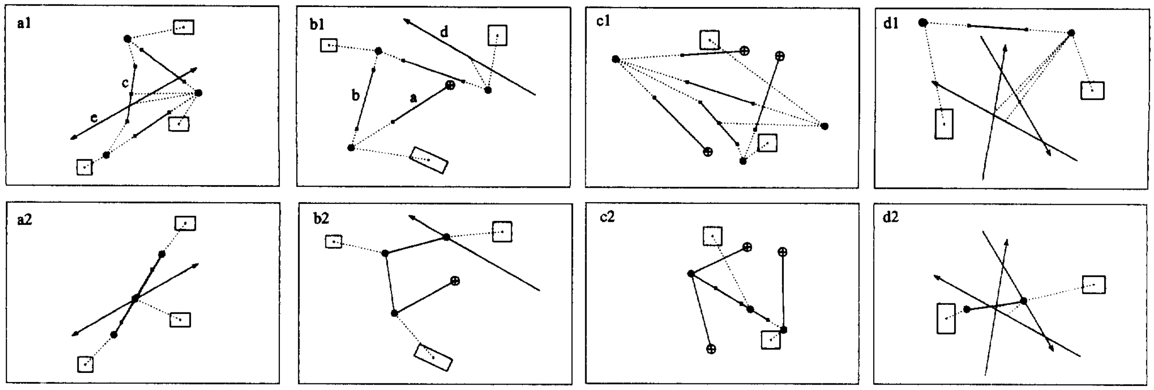

Figure 2: Some examples of constrained models including inequality constraints. Four examples are shown, the original input (measurements, constraints and model points) are shown above and the solution is shown below.

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 line given in Section 6 with the following two changes:
The constraints are now associated with components rather than with single points. Additionally, following the fusion of the constraints associated with component C_k, we find and fuse matched measurements for all the feature points belonging to component C_k. The matching process applied for the feature points 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 on Simulated Data

We applied our method to estimate the pose of a $2D$ constrained model consisting of single point components. We used the parametric modeler described in [7] which we developed based on our techniques. The modeler enables the definition of constraint graphs and the constrained models in Figure 2 were created using this software. In the figures, model points are shown as full circles and measurements are described as rectangles positioned at the measurement location and having width and length proportional to the s.t.d. of the measurement. A ***fixed-location distance*** constraint is visualized by a line segment with one endpoint at the fixed location and the other connected to the constrained point (see constraint 'a' in b1). An ***inter-point distance*** constraint is visualized by a line segment connecting the constrained points (constraint 'b' in b1). A ***three points co-linearity*** constraint is shown by a fixed length line segment connected on both its ends and its middle to the constrained points (constraint 'c' in a1). A ***point on right of line*** constraint (inequality constraint) is shown by an arrow-headed line segment connected to a model point (constraint 'd' in b1) and a ***point on line*** constraint is shown by a double arrow-headed line connected to a model point (constraint 'e' in a1). The connections between constraints and associated model points, are marked by dashed lines. Figure 2 shows four complex examples which include inequality constraints. As can be seen, the solutions conserve the model constraints. Further examples with various initial guesses and different measurements can be found in [6]. It is also shown there that the simulated results obtained by the suggested method indeed correspond to the minimum variance solutions.

9 Results on Real Image Data

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 4,

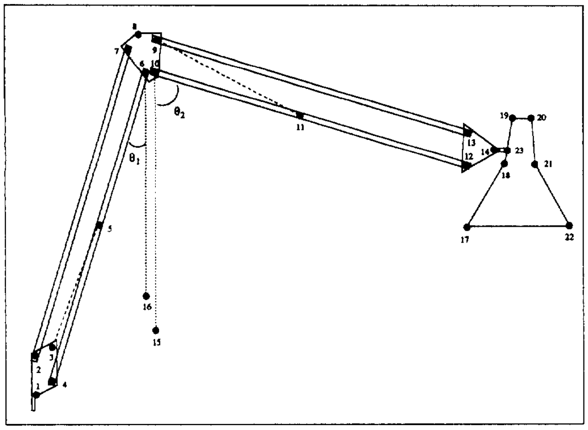

Figure 3: A schematic diagram of a lamp model having 23 points.

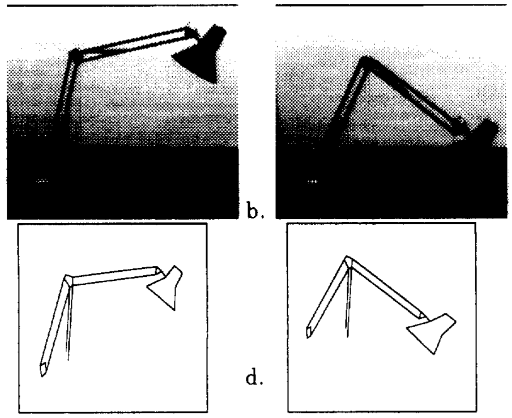

Figure 4: a-b) Images of a desk lamp at different positions. c-d) The corresponding result shown as a synthetic image created from the estimated location vector.

having 5 degrees of freedom. We consider the lamp model as a 23-point model (as shown in Figure 3) and we included the following constraints into the model:

- constant distance constraints between couples of points in the model (e.g. points 10 and 12).
- parallel constraints between 2 pairs of points (between points 2 and 7 and points 4 and 6).
- 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 4a and 4b show 2 examples of lamp images having 2 different positions. Figures 4c and 4d 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.

Additionally, the angles θ_1 and θ_2 (shown in Figure 3) were physically measured in several positions of the lamp. These values were also extracted from the pose estimate obtained with our method. The real values and the constructed values for four typical examples are compared in the following table:

		pose A	pose B	pose C	pose D
Real Values	θ_1	17.0	12.0	14.0	26.0
	θ_2	61.0	103.0	84.0	56.0
No Const.	θ_1	20.2	13.6	13.7	29.3
	θ_2	58.4	104.6	80.2	53.4
With Const.	θ_1	16.5	12.3	14.1	25.8
	θ_2	60.8	104.2	84.7	57.4

The table values show the improvement in the reconstructed angle values when the fusion includes the constraints. The difference between the real angle values and the reconstructed values decreases when the constraints are fused. The s.t.d of these differences is 2.60 for the reconstruction without fusing the constraints and 0.73 for the reconstruction with constraint fusion.

The importance of propagating the pose information of each component to its neighboring components, is shown in Figures 5 and 6. Figure 5 shows several views of a synthesized lamp reconstructed only from the 3D measurements taken on the lamp shown in Figure 4b. Figure 6 shows the same views after mutual information was propagated between the components through the constraints. The improvement is significant, as demonstrated.

9.1 Interpretation Results

Figure 7 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 kth level of this I.T. represents a possible matching between the kth model point, as numbered in Figure

3, and some particular measured point. The measurements are numbered according to their real correspondence (i.e. the true match of the kth measured point is the kth 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.

10 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 constraints are general and can be associated with any number of different parts of the model. The validity of the framework was shown on real and simulated images.

The suggested method has several advantages over existing methods:

1. In any pose estimation method, exploiting the information supplied by the measurement requires a definition of the functional dependence between the measurement and the estimated parameters. In the parametric methods (Section 3), this dependence is not simple since it must include all the parameters on which the measurement depends. Additionally, the order of the nonlinearity of the dependence equations increases with the number of parameters. In our method, the functional dependence includes only the local parameters (i.e. only the parameters that define the transformation of the measured component) since the dependence of the measurement on other parameters is expressed through the constraints of the model. This local dependence is simply defined and is not as highly nonlinear as obtained by the parametric methods. Additionally, there is no need to reduce the constrained parameter space into a set of free parameters (as is performed in the parametric methods), thus the definition of the set of parameters to be estimated, is simple.

2. The information obtained on the position of a given component is propagated to all other components of the model through the constraints between them. Thus the estimated pose of a certain component takes into consideration all the existing measurements and all the defined constraints (this is not true in the divide and conquer methods).

3. 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.

4. 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.

There are computational aspects that were not covered in this paper such as methods to stabilize the convergence, parallelezation technique and methods to speed up the computation and to reduce the time complexity using Optimal Smoothing. This techniques is described in [6].

This framework can be easily extended to handle flexible models made of elastic materials. This extension can be done by associating non-zero uncertainty with the artificial "measurements" produced from the constraints.

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 *Conference on Computer Vision and Pattern Recognition*, pages 461–466, 1991.

[4] R.A. Brooks. Model-based 3-D interpretation of 2-D images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 5:140–150, 1983.

[5] W.E.L. Grimson. On the recognition of parametrized 2-D objects. *International Journal of Computer Vision*, 2:353–372, 1989.

[6] Y. Hel-Or. *Pose Estimation from Uncertain Sensory Data.* PhD thesis, Inst. of Computer Sciense, The Hebrew University of Jerusalem, 1993.

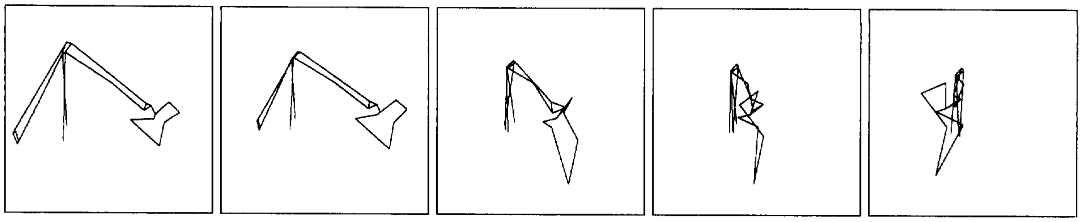

Figure 5: Several views of a synthesized lamp reconstructed from the 3D measurements only.

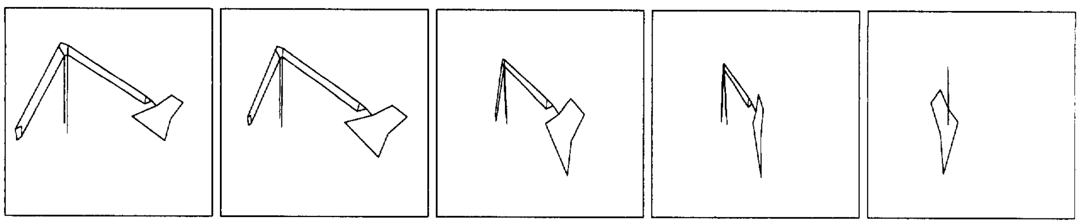

Figure 6: Several views of a synthesized lamp reconstructed from the 3D measurements and the model constraints.

[7] Y. Hel-Or, A. Rappoport, and M. Werman. Relaxed parametric design with probabilistic constraints. In *Solid Modelling-93*, 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 *Conference on Computer Vision and Pattern Recognition*, pages 77–82, 1992.

[10] D.G. Lowe. Fitting parametrized 3-D models to images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13:441–450, May 1991.

[11] P.S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 1. Academic Press, 1979.

[12] 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.

[13] T. Shakunaga. Pose estimation of jointed structures. In *Conference on Computer Vision and Pattern Recognition*, pages 566–572, 1991.

[14] A. Shmuel and M. Werman. Active vision: 3D depth from an image sequence. In *International Conference on Pattern Recognition*, 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.

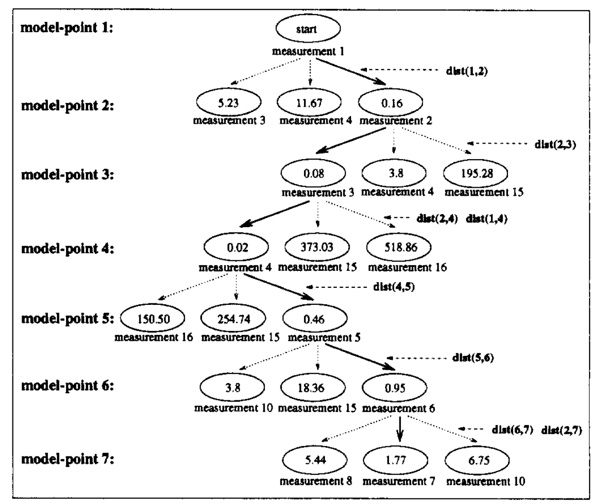

Figure 7: Results of the matching algorithm for the lamp model. A section of the pruned interpretation tree is displayed. Every level of the tree corresponds to one model point, and each node at a particular level corresponds to a possible match between the model point and a measured point. The score 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.