Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Vision and Image Understanding 95 (2004) 105–126
Computer Vision and Image Understanding
www.elsevier.com/locate/cviu

Part-level object recognition using superquadrics

Jaka Krivic* and Franc Solina

Faculty of Computer and Information Science, Computer Vision Laboratory, University of Ljubljana, Tržaška 25, 1000 Ljubljana, Slovenia

Received 18 July 2002; accepted 13 November 2003
Available online 20 March 2004

Abstract

This paper proposes a technique for object recognition using superquadric built models. Superquadrics, which are three-dimensional models suitable for part-level representation of objects, are reconstructed from range images using the recover-and-select paradigm. Using interpretation trees, the presence of an object from the model database can be hypothesized. These hypotheses are verified by projecting and re-fitting the object model to the range image of the scene which at the same time enables a better localization of the object in the scene.

Keywords: Superquadrics; Part-level object modelling; Range images; Object recognition

1. Introduction and motivation

In computer vision many different models have been used for describing various aspects of objects and scenes. Part-level models are one way of representing 3D objects, when particular entities that they describe, correspond to perceptual equivalents of parts. Therefore, several part-level shape models are required to represent an articulated object. Such descriptions are suitable for path planning or manipulation, but they are sometimes not exhaustive enough to represent all the necessary details needed in object recognition.

To obtain part-level description of a scene the image has to be partitioned into segments corresponding to individual parts, and a part model for each of these

* Corresponding author. Fax: +386-1-4264647.
E-mail addresses: jaka.krivic@fri.uni-lj.si (J. Krivic), franc.solina@fri.uni-lj.si (F. Solina).

1077-3142/$ - see front matter
doi:10.1016/j.cviu.2003.11.002

segments has to be recovered. If the two tasks are separated, segmentation does not take into account the shapes that part models can adopt. To avoid this problem, segmentation and recovery can be combined, so that images can only be segmented into parts which are instances of selected part models. To achieve concurrent segmentation and shape recovery, the recover-and-select paradigm can be used.

One of the more popular types of volumetric models are superquadrics [1–5]. They are volumetric models that represent standard geometrical solids as well as shapes in between and are defined by only 11 parameters [6].

In this paper, recognition of structured 3D objects is investigated. Parts of an object form a structure, that distinguishes it from any other object. For the task of recognition of such objects, the relations between parts, the object's structure, are therefore even more important than the shape of the parts itself.

1.1. Segmentation and recovery of superquadrics

Pentland [3] was the first who used superquadrics in the context of computer vision. However, Solina and Bajcsy's method [5] for recovery of superquadrics from pre-segmented range images became more widespread [2].

Several methods for segmentation with superquadrics have been developed. A tight integration of segmentation and model recovery was achieved [7] by combining the "recover-and-select" paradigm [8,9] with the superquadric recovery method [5]. The paradigm works by independently recovering superquadric part models everywhere on the image, and selecting a subset which gives a compact description of the underlying data. *Segmentor* is an object-based implementation of the "recover-and-select" segmentation paradigm using superquadrics and other parametric models [10]. Superquadrics, their mathematical properties, recovery from images and their applications are presented in detail in [2].

1.2. Motivation and related work

The applicability of the *Segmentor* system has been explored in several contexts, in particular for reverse engineering [11]. Segmentation and shape modelling of smooth and regular man-made objects with *Segmentor* is fairly stable, if the objects can be easily represented with superquadric shapes. Segmentation of rough, natural shapes which are not very close to ideal superquadric shapes is less reliable. The superquadric models cannot expand as easily on rough surfaces and complex shapes as on smooth regular objects, which results generally in over-segmentation. Automatic adaptation of the granularity of models to the scale/roughness of the scene is in the context of superquadrics still unresolved [12]. Despite those deficits we decided to test the applicability of the *Segmentor* system for object recognition of articulated objects in complex scenes.

Our aim was to investigate the possible use of part-level descriptions obtained by the *Segmentor* system for recognition of articulated objects. We hypothesized that the configuration of parts and their rough shape should provide enough constraints for successful matching with the models of known objects. The recognition system

would search for matches between scene and model parts, a procedure known as *model-based matching*. The object hypotheses can be subsequently verified by fitting the object model directly to the range data (Fig. 1). Such recognized objects could be further used for higher level reasoning, such as developed by Chella et al. [13]. As a means for scene understanding they used the notion of conceptual space, to link between subconceptual information (in the form of superquadrics) and symbolically organized knowledge.

Superquadrics have been used in several computer vision systems. Raja and Jain [4] tried to relate superquadrics and geons, part primitives introduced by Biederman

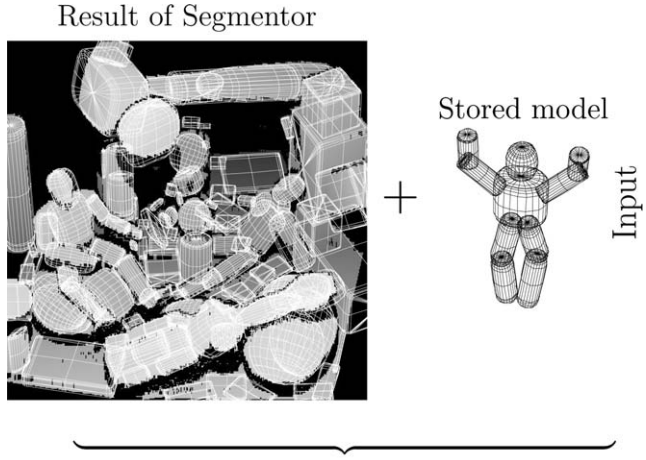

Fig. 1. Overview of the object recognition system.

[14]. They investigated recognition of geons from superquadrics fitted to range data, but did not deal with object made of those parts. Dickinson et al. [1] used superquadrics as modelling primitives to construct objects. The recognition is based on aspects, which are used to model the superquadric parts. Aspects are recovered from an image, and aspect hierarchy is used to infer a set of volumetric primitives and their connectivity relations. The verification of object hypothesis is then basically topological verification of the recovered graph.

Since superquadrics are part-level descriptions, an object recognition system that searches for matches between parts in the scene and parts of the modelled object can be used [1]. One of the first such methods by Nevatia and Binford [15] uses a relational graph structure to represent an object. The recognition then becomes a matter of matching two graphs. The 3DPO vision system developed by Bolles and Horaud [16] uses a "local feature focus" method for constraining the size of the solution search space. Kim and Kak [17] used bipartite matching for fast rejection of inapplicable models, and a combination of bipartite matching and discrete relaxation to prune the possible object hypotheses. Grimson [18] developed the "interpretation tree" method. He arranged all possible matches of scene part with model part in a tree structure—an interpretation tree. The problem of recognition is to find consistent interpretations without exploring all possible ways of matching the scene and model parts, which was done using geometric constraints. A nice introduction to interpretation trees for use in computer vision can be found in [19].

2. Part-level object recognition system

The output of the *Segmentor* system is a set of recovered superquadrics, which represents the parts of the input scene. On this set the search for feasible interpretations of the stored model is employed. When an interpretation is found, it can be verified by projecting the object model into the scene.

The system we propose consists of the following three steps:

(1) perform range image segmentation and superquadric recovery using the *Segmentor*, resulting in a set of N superquadric parts,
(2) search for feasible interpretations of the selected model in the set of N reconstructed superquadrics, based on part by part match using interpretation trees, and
(3) verify hypothesized interpretation by using the structural properties of models and by projecting object models into the range image of the scene.

The second and third steps can be interleaved, to quickly eliminate wrong hypotheses.

2.1. Object model

If an object is to be recognized by the system, the system must have a model of the object. In the proposed system the object is modelled on two levels. On the first level, object's parts are modelled with superquadrics that define the part's size and shape,

such as the superquadrics in Fig. 2B (see also Fig. 8B). On the second level the part structure is described by defining the connections between parts, such as connections in Fig. 2B (see also Fig. 8B). One part is given the central role in the object's model. To the central part the object position and general orientation can be assigned. Other parts are connected to their "parents parts" by a joint. Vector $\mathbf{r}_{ij}$ denotes the position of joint connecting parts i and j relative to the center of part i. Therefore, to define a joint two vectors $\mathbf{r}_{ij}$ and $\mathbf{r}_{ji}$ are needed.

There are two types of joints: rigid and flexible. Rigid joints contain besides positional parameters, predefined (constant) rotational parameters, denoted by rotational matrix $\mathbf{R}_i$, and therefore rigidly 'glue' the two parts together. The object in Fig. 2, for example, contains two rigid joints. Flexible joints, however, do not have fixed rotational parameters, but can be assigned any value from a given interval for rotating the connected parts into the right configuration. Such flexible joints connect parts of non-rigid objects such as the figurine in Fig. 8. Of course, the values of rotational parameters could also be constrained, as, for example, would be the case of a human arm [20], which can only move in certain ways, but this is beyond the scope of our work.

In this paper, $\mathbf{r}_{ij}$ stands for the joint position connecting parts i and j, as described above, $\mathbf{c}_i$ is the center of the superquadric that matches, or should match part i, $\mathbf{R}$ is a ZYZ rotation matrix, and ϕ_i, θ_i, and ψ_i are rotation angles for part i.

Since we focused our work on the recognition phase, we built the models manually by measuring the parts and approximating the superquadric and other parameters for each part and joint (e.g., Fig. 2).

2.2. Superquadrics

Superquadrics are a family of volumetric models, which were first introduced in computer graphics by Barr [6] and later gained popularity in computer vision [1–5,7]. Basic superquadric shapes are compact representation of 3D shapes as they

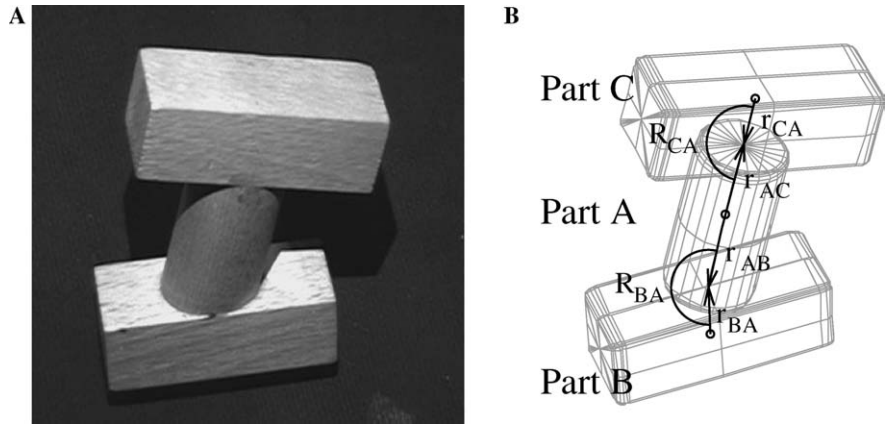

Fig. 2. Simple object (A) and its model (B).

are described by only 11 parameters ($a_1,\ a_2,\ a_3$ [size], $\epsilon_1,\ \epsilon_2$ [shape], $t_x,\ t_y,\ t_z$ [translation], $\phi,\ \theta,\ \psi$ [rotation]). The surface of a superquadric in local coordinate frame is defined by

$$\mathbf{s}(\eta,\omega)=\begin{bmatrix}s_x\\ s_y\\ s_z\end{bmatrix}=\begin{bmatrix}a_1\cos^{\epsilon_1}\eta\cos^{\epsilon_2}\omega\\ a_2\cos^{\epsilon_1}\eta\sin^{\epsilon_2}\omega\\ a_3\sin^{\epsilon_1}\eta\end{bmatrix},\quad \begin{matrix}-\frac{\pi}{2}\leqslant\eta\leqslant\frac{\pi}{2},\\ -\pi\leqslant\omega<\pi.\end{matrix} \tag{1}$$

From this equation superquadric implicit function can be derived:

$$F(x,y,z)=\left(\left(\frac{x}{a_1}\right)^{\frac{2}{\epsilon_2}}+\left(\frac{y}{a_2}\right)^{\frac{2}{\epsilon_2}}\right)^{\frac{\epsilon_2}{\epsilon_1}}+\left(\frac{z}{a_3}\right)^{\frac{2}{\epsilon_1}} \tag{2}$$

with superquadric surface points satisfying the equation $F(x,y,z)=1$. Fig. 3 shows some superquadric shapes.

2.3. Superquadric recovery with the segmentor system

Let us briefly describe the Segmentor system [2,10] for range image segmentation and superquadric recovery. The system uses the *recover-and-select* paradigm [9] in the segmentation process. The input to the system is a range image, captured by the range scanner in our lab (see Section 3 for the setup). In the first step, initial (seed) descriptions are placed everywhere on the range image (see Fig. 4A). A description consists of a set of range points and a corresponding model, in this case a superquadric model. The next step is a growing stage (compare Figs. 4A–D as models grow in size). To each description new points are added, that are close to the model, and a new model is reconstructed on this extended set of range points. After several growing stages, several models may completely or partially overlap. That is the moment when the selection takes place (compare Figs. 4A–D as the number of models decreases). Using the *minimum description length* criterion a subset of descriptions are selected, that produce the simplest description of the range image. Growing and selection stage can be interleaved in order to speed-up the process.

2.4. Model matching

The output of the Segmentor system is therefore a set of N superquadrics, which compose the scene. We will call them scene parts. One can easily imagine the process

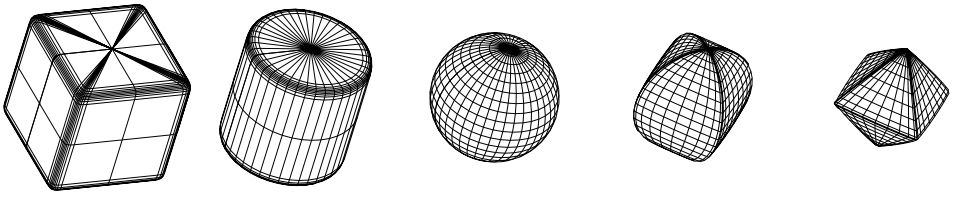

Fig. 3. Some basic superquadric shapes.

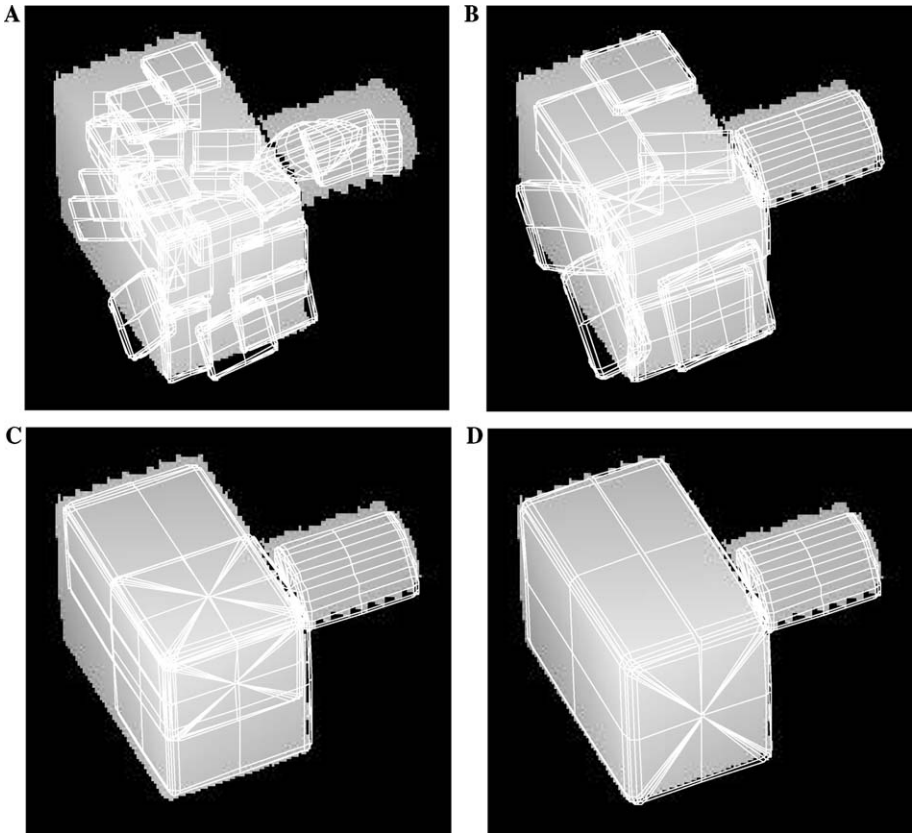

Fig. 4. Superquadric recovery with the Segmentor system: (A) placed superquadric seeds, (B) after two growing stages and selection, (C) after 6 growing stages and selection, and (D) after 14 growing stages and selection.

of recognizing an object as matching scene parts with part models of the stored body model. All possible matches arranged in a tree structure are called an *interpretation tree* [18]. Nodes in an interpretation tree represent a match between a part of the scene and a part of the model. The search for correct interpretation begins at the root of the interpretation tree. The root expands to all possible matches for the first model part. At the first level of interpretation tree search, every one of N scene parts is therefore matched to the first model part. From a given node, the search continues in depth only if the match represented by that node is consistent, i.e., if the two parts represented by that node are similar. On a given level of the interpretation tree search, the corresponding model part is matched to all scene parts from the set of N parts, except the ones that have already been matched on some higher levels of the tree.

In depth-first search, which was used in our system to examine the interpretation tree, the order of model parts as they pertain to the depth of the tree can be

important for finding an interpretation quickly. The parts that are reconstructed more consistently and from more viewpoints should be closer to the tree root. The system thus searches more probable interpretations first. One can, of course, easily implement many other enhancements, such as tree pruning, where two parts are matched only if they are in the right distance to the parts already matched. Best-first search could also be implemented by sorting the matches based on part similarity.

In real scenes some parts of an object may be hidden to the viewer and some occluded by other objects or parts. Also, the part detector can miss some parts or introduce some spurious ones. To enable the system to deal with such cases, a fictitious scene part that matches every model part is introduced. A match between this fictitious scene part and a model part is called a *wildcard match* and is simply appended to the list of scene parts.

When the search through the interpretation tree reaches a leaf one gets a consistent interpretation. But because the constraints involved in match consistency test are local in nature, the interpretation does not have to make sense globally. In general, there is no guarantee that a found interpretation makes global sense. These interpretations must therefore be taken only as hypotheses. For most problems one can come up with a test for global consistency which discards wrong hypotheses, a process called *interpretation verification*.

Algorithm 1 (see Appendix A) outlines the model matching procedure used in the system.

2.5. Match consistency

As mentioned above, if an object is present in some scene, it should consist of the same parts as the object's model. In reality, of course, the parts are not exactly the same, but should be similar enough. The comparison between two superquadric parts, should therefore be tolerant to slight (or great) changes in part shape and size. Superquadric parameters cannot be used directly for comparison of two superquadrics because several sets of parameters can lead to the same size and shape of a part [2]. Therefore when comparing scene part f_j with model part m_i, a set of constraints $\mathcal{T}_i$ is used to determine the part similarity (i.e., match consistency), which is dependent on superquadric recovery on the model part m_i. In this paper, (m_i, s_j) denotes a match between model part i and scene part j. A consistent match (m_i, s_j) (where matches for parts m_0 to m_{i-1} are consistent) means, that the search can continue with the next match (m_{i+1}, s_j) at next level $i+1$ of the interpretation tree, whereas an inconsistent match (m_i, s_j) stops further search in depth and continues with the next match (m_i, s_k) on the same level i of the interpretation tree.

A basic constraint, that can be included in every part's constraint set $\mathcal{T}_i$, is a volume constraint $\mathcal{V}_i(V) : V \in [V_i^{\min}, V_i^{\max}]$. If a volume V of a scene part is within the model's part interval, $V_i^{\min} \leqslant V \leqslant V_i^{\max}$, the two parts represent a possible match.

Superquadric recovery on some shapes is not reliable and produces many (two or more) overlapping superquadrics, that correspond to a single model part. In those cases, the volume constraint can be extended to $\mathcal{V}'_i(V) : (\exists \mathrm{sq} \in S_i \wedge (\sum_{\mathrm{center(sq)} \in S_i} V_{\mathrm{sq}}) \in [V_i'^{\min}, V_i'^{\max}]) \vee (\overline{\exists} \mathrm{sq} \in S_i \wedge V \in [V_i^{\min}, V_i^{\max}])$, as follows:

- if there are any superquadrics, whose centers are less than some distance S_i from the center of the considered superquadric, the sum of their volumes $\sum V_{\text{sq}}$ (including the volume of the considered superquadric) should be in the predefined interval $V_i'^{\min} \leqslant \sum V_{\text{sq}} \leqslant V_i'^{\max}$. Distance S_i can be assigned a value of the perimeter of the largest sphere, that can fit into the model part being matched.
- if there are no other superquadrics at such a distance, the part's volume V should be in the usual interval $V_i^{\min} \leqslant V \leqslant V_i^{\max}$.

The two cases are dealt with separately (with different values), because the shared space of the superquadrics is not taken into account.

For parts with reliable superquadric reconstruction, such as the parts of the object in Fig. 2, size and shape along minimal inertia axis can be used. The size constraint is defined as $\mathcal{S}_{i,l}(a_l') : a_l' \in [a_{i,l}^{\min}, a_{i,l}^{\max}]$, $l = 1, 2, 3$. Similarly, the shape constraint is defined as $\mathcal{H}_{i,m}(\epsilon') : \epsilon_m' \in [\epsilon_{i,m}^{\min}, \epsilon_{i,m}^{\max}]$, $m = 1, 2$. Constraints can be computed as follows: first, inertial moments along x, y, z axes are computed for part f_j, and sorted. Next, a_l' is assigned the a_k parameter that corresponds to the lth lowest inertial moment value (e.g., when inertial moment along y axis is lowest, $a_1' := a_2$, since a_2 is the size along y axis). ϵ_1' is assigned the ϵ_1 parameter when inertial moment along x or y axis is smallest, and ϵ_2 when inertial moment along z is smallest. ϵ_2' is assigned the remaining ϵ parameter. For parts with a very reliable reconstruction the volume difference [21] constraint could be used in order to match parts to shape and size as accurately as needed. The volume difference constraint was not implemented in our system though, because of its high time complexity.

Note that the properties used above are unary. When including a scene part in an interpretation, there are possibly other parts already included. In order to reduce the search space, binary (n-ary) constraints, such as distance between two parts, can be introduced into the part matching procedure. The distance constraint $\mathcal{D}_{i,j}(d) : d \in [d_{i,j}^{\min}, d_{i,j}^{\max}]$ is based on distance d between the centers of scene superquadrics that represent parts m_i and m_j.

The purpose of part match consistency is to prune the interpretation tree, leading to faster interpretation discovery. The constraints involved should be adjusted so that the part matching procedure rejects as many unsuitable parts, while accepting any possible part matches that may appear in scene reconstructions. In this way the system does not "overlook" any objects and finds them quickly.

2.6. Interpretation verification

Interpretation verification means that the system should answer the question: "Does the given set of parts really represent the object X?"

First, the system can reject interpretations that include too many wildcard matches, by setting a threshold P on the real interpretation size. For example, if the object is a part of a fence with twenty iron poles welded to a frame (24 parts altogether), the object could be recognized even if 16 poles are missing, thus a threshold $P = 8$ parts (one-third of model parts) would be reasonable. On the other hand when recognizing the figurine from Fig. 8, three matched parts (approximately one0third of model parts) does not necessarily indicate the object's

presence. One would expect at least five matched parts to be sure of the object's presence. By rejecting interpretations that include too few real matches, the system may therefore reject some correct interpretations (false negatives), but it will reject many more wrong ones (false positives), since the probability that some parts will randomly form a structure similar to the structure of the object decreases as number of matched parts increases. The threshold P can be set to some fraction of the number of model parts and depends on the modelled object as well as on the application. For most objects the threshold P can be set to around half of the number of model parts.

Second, for a given interpretation, the hypothetical object position and part configuration can be computed. We focused our work on articulated objects consisting of elongated parts with unreliable reconstructions, that are joined near the longer ends. Using that assumption, the configuration can be computed efficiently. Let the part's main axis be the axis of minimal inertia [2,22]. Our analysis of such objects showed that the main axes of scene parts are well aligned with true main axes of the object model parts. When a joint is configured so that it connects two parts, the following rotation of the subordinate part is the rotation that aligns its main axis with the main axis of the matched scene part:

$$\begin{aligned}
\mathbf{R}_{\mathbf{X}\to\mathbf{s}}: &\quad \phi = \arctan\tfrac{-s_x}{s_y}, &\quad \theta = \tfrac{\pi}{2}, &\quad \psi = \arctan\tfrac{\sqrt{s_x^2+s_y^2}}{s_z},\\
\mathbf{R}_{\mathbf{Y}\to\mathbf{s}}: &\quad \phi = 0, &\quad \theta = \arctan\tfrac{s_z}{-s_x}, &\quad \psi = \arctan\tfrac{\sqrt{s_x^2+s_z^2}}{s_y},\\
\mathbf{R}_{\mathbf{Z}\to\mathbf{s}}: &\quad \phi = \arctan\tfrac{s_y}{s_x}, &\quad \theta = \arctan\tfrac{\sqrt{s_x^2+s_y^2}}{s_z}, &\quad \psi = 0,
\end{aligned} \tag{3}$$

where $\mathbf{R}_{\ldots}$ is a ZYZ rotation matrix, ϕ, θ, and ψ are rotation parameters and $\mathbf{s} = [s_x, s_y, s_z]^{\mathrm{T}}$ is a scene part's unit main axis vector rotated in the local coordinate frame of the part superior to the part being configured. In the general case, the object's configuration is hard to determine due to inherent rotational ambiguity of superquadrics. For computing the configuration, custom procedures tailored to particular objects, or types of objects, would have to be developed.

After an object model is approximately configured to its interpretation, this configuration can serve as the basis for the third step in the interpretation verification. The individual superquadric part of the object model can then be fitted to those regions in the range image that correspond to their position given by the approximated configuration. To fit individual superquadric models to such part regions the standard fitting method was used [5]. The fitting function [2,10]

$$G(\Lambda) = a_1 a_2 a_3 \sum_{i=1}^{N} \left(F^{\epsilon_1}(x_i, y_i, z_i) - 1\right)^2, \tag{4}$$

where F is the superquadric implicit function from Eq. (2) and $[x_i, y_i, z_i]^{\mathrm{T}}$ the point i from the range image region was minimized only for the position and orientation parameters, i.e., $\Lambda = (t_x, t_y, t_z, \phi, \theta, \psi)$, while the size (a_1, a_2, a_3) and shape (ϵ_1, ϵ_2) parameters were fixed to the values of the tested model part superquadric. The position and orientation parameters of the tested superquadric were used in the

minimization as initial parameters. For model parts with reliable reconstructions, the interpretation is rejected, if the error of the fitting function [2,10] is greater than threshold $E = 2.5$ (the same as used in range image segmentation). For parts with unreliable reconstruction, the model superquadric is fitted in the same way, but the interpretation is rejected if the poles (points where the main axis pierces the superquadric surface) move more than a threshold D_i.

The final interpretation therefore consists of the object model whose configuration in 3D has been refined by fitting each superquadric of the object model to its corresponding region in the range image.

Rigid joints are then further verified for consistency. Due to the rotational ambiguity of superquadrics, we did not deal thoroughly with joint rigidity, but rather compared the angle between the two main axes.

There is another aspect of interpretation verification, namely the feasibility of the object's configuration. If an articulated object is set by the interpretation process into a non-feasible configuration, the verification process should reject it. This paper does not deal with configuration feasibility, but this could be applied to the system presented by defining sets of valid intervals for joint rotation parameters. The joint rotation parameters extracted from the final interpretation would then be compared to those sets thus determining if self-penetration and other non-feasible poses have occured.

2.7. A simple example

Let us look at a simple example of interpretation search in a greater detail. The input scene and its range image are depicted in Fig. 5, in which we search for the simple test object seen in Figs. 2 and 6B. The parts of the test object are labelled A, B, and C. The first step of the recognition process is superquadric recovery using the Segmentor system. The superquadric reconstruction on scenes such as the one in Fig. 5, where all parts can be perfectly modelled by superquadrics, is very reliable. For each scene part one superquadric is reconstructed, which describes the corresponding scene part very well, and there is almost no overlapping. The result can be seen in Fig. 6A. Parts in the reconstruction are labelled 0–5. The threshold P on real interpretation size is set to 2 (the interpretation must include at least 2 parts).

Next, the search for possible interpretations begins using interpretation trees. The interpretation tree for finding the test object is shown in Fig. 7. The search begins at the root. The root expands into nodes representing matches between model part A and scene parts from 0 to 5. First, part A is compared to part 0. Since part A is a cylinder and part 0 is a block, the match is not consistent and the search does not continue in depth. It rather proceeds on the same level by visiting the right sister node and comparing parts A and 1. Again, this match is discarded due to part shape mismatch. Visiting right sister node on the same level, the search continues by comparing part A with part 2. Size and shape approximately match, so that the search can continue in depth, by searching for a match for model part B. The comparison to scene part 0 delivers a consistent match, since the size and shape match. Continuing one level deeper, matches for model part C are searched. The first node yields a

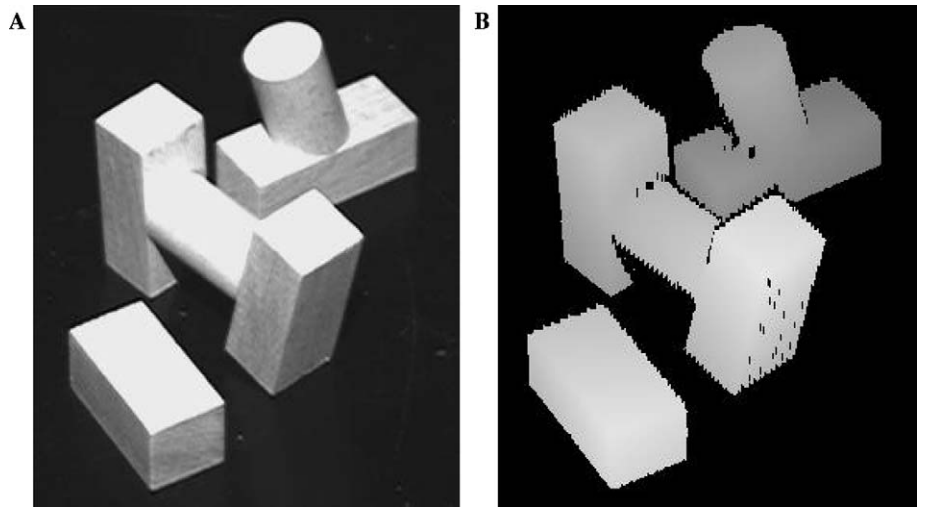

Fig. 5. Simple scene (A) containing the object from Fig. 2 and the corresponding range image (B).

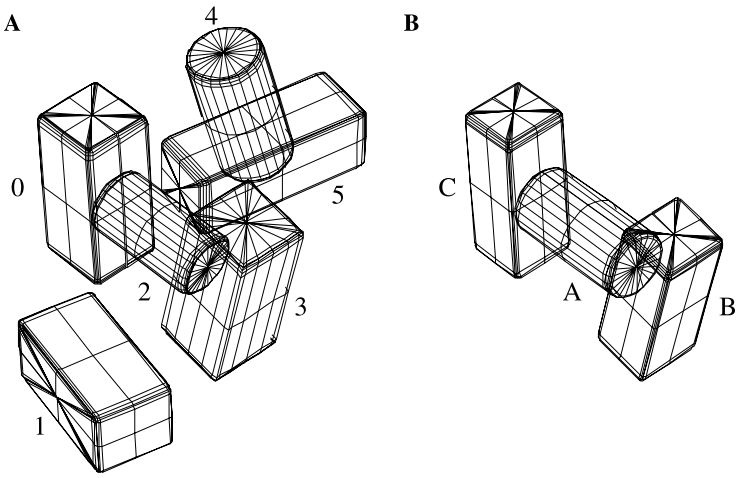

Fig. 6. (A) Superquadric reconstruction of the scene from Fig. 5 and (B) labelled model from interpretation verification.

consistent match between parts C and 1. Since this is a leaf node, a consistent interpretation $(A, B, C) = (2, 0, 1)$ is obtained.

Although the parts have pairwise the same size and shape, a glance at Fig. 6 can tell that the recovered parts $(2, 0, 1)$ do not represent the object in question, since their configuration is wrong. This is why every consistent interpretation derived by the interpretation tree must be verified using the properties of the whole object. The parts in the interpretation should conform to the same structure as parts that

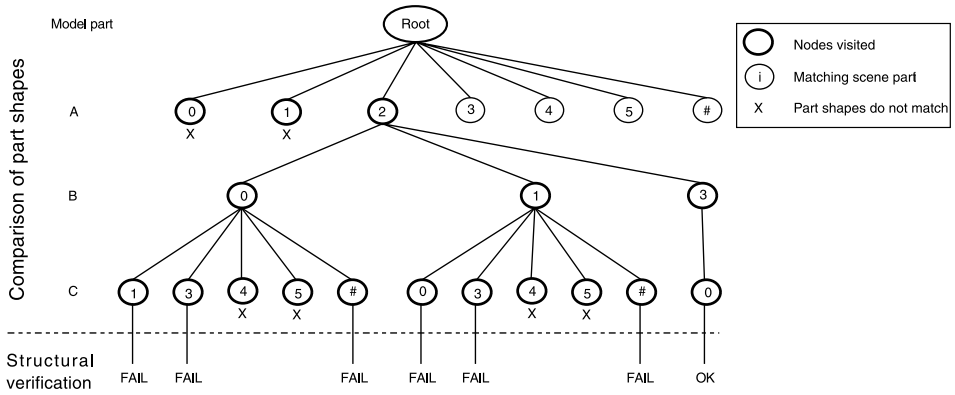

Fig. 7. Interpretation tree for scene in Fig. 5.

compose the model. The system can verify if the configuration occupied by the scene parts in the interpretation corresponds to the model using structural information described in Section 2.1.

The first step in the process of verifying the interpretation $(A, B, C) = (2, 0, 1)$ is putting a threshold P on its size. Since the interpretation includes three real matches, it passes the first test. Next, the configuration of the hypothesized object is compared to the object's model. Scene part labelled 1 is too distant from part 2, so the interpretation is rejected, and the search continues at the right sister node, with interpretation $(A, B, C) = (2, 0, 3)$. When comparing the hypothesized configuration with the model, the distances between part centers match. But since the joints in the object are rigid, the joint rotations of the hypothesized object do not match the modelled ones, because the model parts A and B are slightly tilted whereas scene parts 2 and 0 are perpendicular. If both joints were flexible, the configuration would match, the rotational parameters would be computed, and the superquadric part fitting would proceed. Since the hypothesized object's configuration would be accurate, the superquadric parts of the model would not move or rotate much in the process of fitting, and the interpretation would succeed.

Let us skip forward in the interpretation tree search until the interpretation $(A, B, C) = (2, 0, \#)$ is found. After trying all possible matches for model part C, there is also a possibility, that the part in question is missing (is occluded, or the reconstruction is not appropriate). The hash sign (#) in the interpretation stands for a wildcard, that is a fictitious part that matches every model part. A wildcard match is simply appended (at the end) of the list of scene parts. Interpretation $(A, B, C) = (2, 0, \#)$ is thus consistent, but fails again on structural verification. For the purpose of demonstration, let us imagine that part 0 is occluded from the scene. The interpretation tree search would then lead to interpretation $(A, B, C) = (2, 3, \#)$, which is structurally sound, and also represents the best interpretation for the scene.

The search continues with three more consistent interpretations, which fail on structural verification, until the correct interpretation $(A, B, C) = (2, 3, 0)$ is found.

3. Experiments

We decided to use human figures as generic articulated test objects for our recognition task. We were not interested in the specific problem of modelling the human form and do not want to compete with dedicated human form capture systems, although it should be mentioned that systems using superquadrics for modelling humans do exist [23].

Since the work space of our range scanner is rather small (see next subsection), we decided to use toy figurines instead. We selected figurines representing "Commander Data" from the Star Trek series (Fig. 8A). Their arms and legs are flexible and the figurines can thus be configured into many different poses.

3.1. Experimental setup

The experimental setup for the system was as follows: range images were obtained by the structured light range scanner from our lab. Its main components are an ABW LCD projector for projecting the structured light sequence onto the scene, a Sony XC-75CE camera for capturing the image sequence, and Linux based software [24] that controls the projector and camera, and generates the range image from the captured sequence. A range image is an array of 450×450 elements signifying the distance between the element and the camera. The work space of the scanner is about $25 \times 25 \times 20$ cm, so that objects larger than that can not be scanned as a whole. It takes the scanner about ten seconds to capture a range image.

Captured range images were processed with the *Segmentor* system [2,10]. On a 400 MHz PC, the processing took from 1:30 (simple scenes) to 3:00 h (complex scenes).

The resulting sets of superquadric descriptions were processed with the recognition system as described in the previous sections. The system was implemented in C++, and the processing times for some examples can be seen on Table 1. Tables

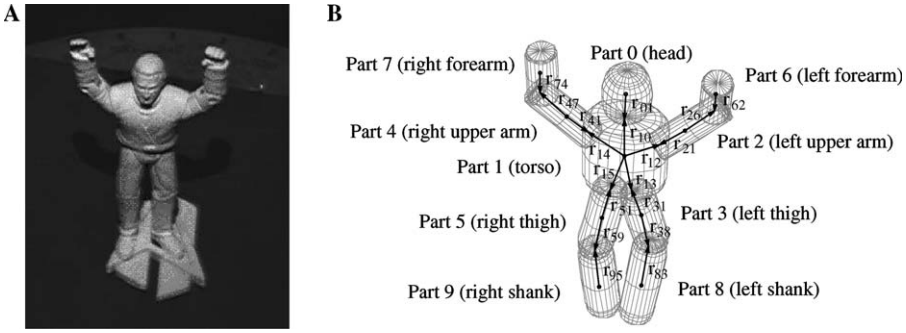

Fig. 8. Toy figurine (A) is modelled in two levels (B): superquadric part models define the size and shape of individual parts (grey models) while the structural level (vectors $\mathbf{r}_{ij}$) defines how parts are connected to each other.

Table 1
Processing times for some input sets

Input	Model parts	Scene parts	Objects	Comp. time (s)
H-object 1, Figs. 5 and 6	3	6	1	2.9
H-object 2, Fig. 9	3	8	1	3.4
Toy figurine 1, Fig. 11	10	183	2	48.7
Toy figurine 2	10	121	2	42.5

3 and 4 show constraints and verification parameters' values that were used in the experiments, respectively.

We built the model of the figurine manually. The model consists of superquadrics (Fig. 8B). Each superquadric represents one of the major body parts: head, torso, a pair of upper arms and forearms, and a pair of thighs and shanks. Due to the limited scale of parts which can be recovered on the selected range image resolution by the *Segmentor*, the model does not include distinct models of hands and feet. Each body part is described by a superquadric of a particular size and shape. The torso is given a central role in the model. The head and upper arms and legs are attached to it via joints (Fig. 8B). For each of those parts the joint position in relation to the center of the part itself ($\mathbf{r}_{i1}$) and to the center of the torso ($\mathbf{r}_{1i}$) is defined. Similar is true for lower extremities. The parameter values for all parts were obtained by measuring the figurine and are listed in Table 2.

The figurine is interesting in several ways. It is fairly realistic and naturally shaped and therefore cannot be perfectly modelled by superquadrics. Since the surfaces are not smooth, the reconstruction of superquadrics on their range images is less stable. There can be several superquadrics reconstructed on a single scene (object) part, or a single superquadric can span over several scene (object) parts. The flexibility of body joints makes the matching problem even more complex than if the object part configuration would be rigid.

Table 2
Model parameters for toy figurine object from Fig. 8

No.	Part	a_1	a_2	a_3	ϵ_1	ϵ_2	Volume
0	Head	8	8	10	0.7	1.0	3185
1	Torso	14	10	15	0.3	0.9	13077
2, 4	Upper arm_x	5	5	13	0.1	1.0	2027
3, 5	Thigh_x	7	7	17	0.3	1.0	4995
6, 7	Forearm_x	5	5	10	0.1	1.0	1559
8, 9	Shank_x	6	6	17	0.3	1.0	3670

Joint positions

$\mathbf{r}_{10} = [0,0,15]^{\mathrm{T}},\ \mathbf{r}_{01} = [0,0,-10]^{\mathrm{T}},\ \mathbf{r}_{12} = [15,0,8]^{\mathrm{T}},\ \mathbf{r}_{21} = [0,0,7]^{\mathrm{T}},$
$\mathbf{r}_{13} = [5,0,-22]^{\mathrm{T}},\ \mathbf{r}_{31} = [0,0,7]^{\mathrm{T}},\ \mathbf{r}_{14} = [-15,0,8]^{\mathrm{T}},\ \mathbf{r}_{41} = [0,0,7]^{\mathrm{T}},$
$\mathbf{r}_{15} = [-5,0,-22]^{\mathrm{T}},\ \mathbf{r}_{51} = [0,0,7]^{\mathrm{T}},\ \mathbf{r}_{26} = \mathbf{r}_{47} = [0,0,-9]^{\mathrm{T}},$
$\mathbf{r}_{62} = \mathbf{r}_{74} = [0,0,10]^{\mathrm{T}},\ \mathbf{r}_{38} = \mathbf{r}_{59} = [0,0,-14]^{\mathrm{T}},\ \mathbf{r}_{83} = \mathbf{r}_{95} == [0,0,17]^{\mathrm{T}}$

$x = \{1,2\}$.

3.2. Constraint values and verification parameters

Reconstructions of superquadrics on range images of the object taken in different poses and from different viewpoints differ greatly. The exception is the head since the analysis of superquadric reconstructions of the human body showed that the head was the most consistently reconstructed body part. At the same time, the head is also the only part that does not change significantly its relative position in relation to the torso. It was therefore reasonable for the head part to use in part matching beside the volume constraint also the size $\mathcal{S}$ and the shape $\mathcal{H}$ constraints, to early on reject as many unsuitable parts as possible.

Superquadrics reconstructed on the torso region differ the most from the torso's model superquadric. On this region several possibly overlapping superquadrics can be recovered, which can partially extend even into regions belonging to extremities. Thus, the extended volume constraint $\mathcal{V}'$ was used for the torso part.

Table 3 lists the constraint values, while Table 4 lists verification parameters used for the figurine object. Values were defined on the basis of thirty superquadric reconstructions of the object's range images.

4. Results

Let us first present an example recognition of the simple object from Fig. 2. Figs. 9A–D show the scene, the superquadric reconstruction of the scene, the best hypothesized interpretation and the verified interpretation, respectively. The interpretation

Table 3
Match consistency test values for the toy figurine object from Fig. 8

No.	Part	Constraints	Constraint values
0	Head	$\mathcal{T}_0 = \{\mathcal{V}_0, \mathcal{S}_{0,1}, \mathcal{S}_{0,2}, \mathcal{S}_{0,3}, \mathcal{H}_{0,1}, \mathcal{H}_{0,2}\}$	$V_0^{\min} = 1000$, $V_0^{\max} = 6000$, $a_{0,1}^{\min} = 6.5$, $a_{0,1}^{\max} = 15$, $a_{0,2}^{\min} = 5$, $a_{0,2}^{\max} = 11$, $a_{0,3}^{\min} = 3$, $a_{0,3}^{\max} = 9$, $\epsilon_{i,1}^{\min} = 0.5$, $\epsilon_{i,1}^{\max} = 1.4$, $\epsilon_{i,2}^{\min} = 0.4$, $\epsilon_{i,2}^{\max} = 1.5$
1	Torso	$\mathcal{T}_1 = \{\mathcal{V}'_1, \mathcal{D}_{1,0}\}$	$V_1^{\min} = 3500$, $V_1^{\max} = 14000$, $V_1'^{\min} = 5500$, $V_1'^{\max} = 20000$, $S_1 = 12$, $d_{1,0}^{\min} = 16, d_{1,0}^{\max} = 27$
2, 4	Upper arm	$\mathcal{T}_{(2\|4)} = \{\mathcal{V}_{(2\|4)}, \mathcal{D}_{(2\|4),1}\}$	$V_{(2\|4)}^{\min} = 300$, $V_{(2\|4)}^{\max} = 6500$ $d_{(2\|4),1}^{\min} = 14, d_{(2\|4),1}^{\max} = 31$
3, 5	Thigh	$\mathcal{T}_{(3\|5)} = \{\mathcal{V}_{(3\|5)}, \mathcal{D}_{(3\|5),1}\}$	$V_{(3\|5)}^{\min} = 500$, $V_{(3\|5)}^{\max} = 8000$ $d_{(3\|5),1}^{\min} = 20, d_{(3\|5),1}^{\max} = 34$
6, 7	Forearm	$\mathcal{T}_{(6\|7)} = \{\mathcal{V}_{(6\|7)}, \mathcal{D}_{(6\|7),(2\|4)}\}$	$V_{(6\|7)}^{\min} = 300$, $V_{(6\|7)}^{\max} = 3500$ $d_{(6\|7),(2\|4)}^{\min} = 7, d_{(6\|7),(2\|4)}^{\max} = 23$
8, 9	Shank	$\mathcal{T}_{(8\|9)} = \{\mathcal{V}_{(8\|9)}, \mathcal{D}_{(8\|9),(3\|5)}\}$	$V_{(8\|9)}^{\min} = 300$, $V_{(8\|9)}^{\max} = 4000$ $d_{(8\|9),(3\|5)}^{\min} = 9, d_{(8\|9),(3\|5)}^{\max} = 38$

Table 4
Values of interpretation verification parameters used in the experiments

Meaning	Variable	Value
Interpretation size threshold	P	5
Maximum pole rotation thresholds	D_1	3
	D_2, D_3, D_4, D_5	11
	D_6, D_7	6
	D_8, D_9	7

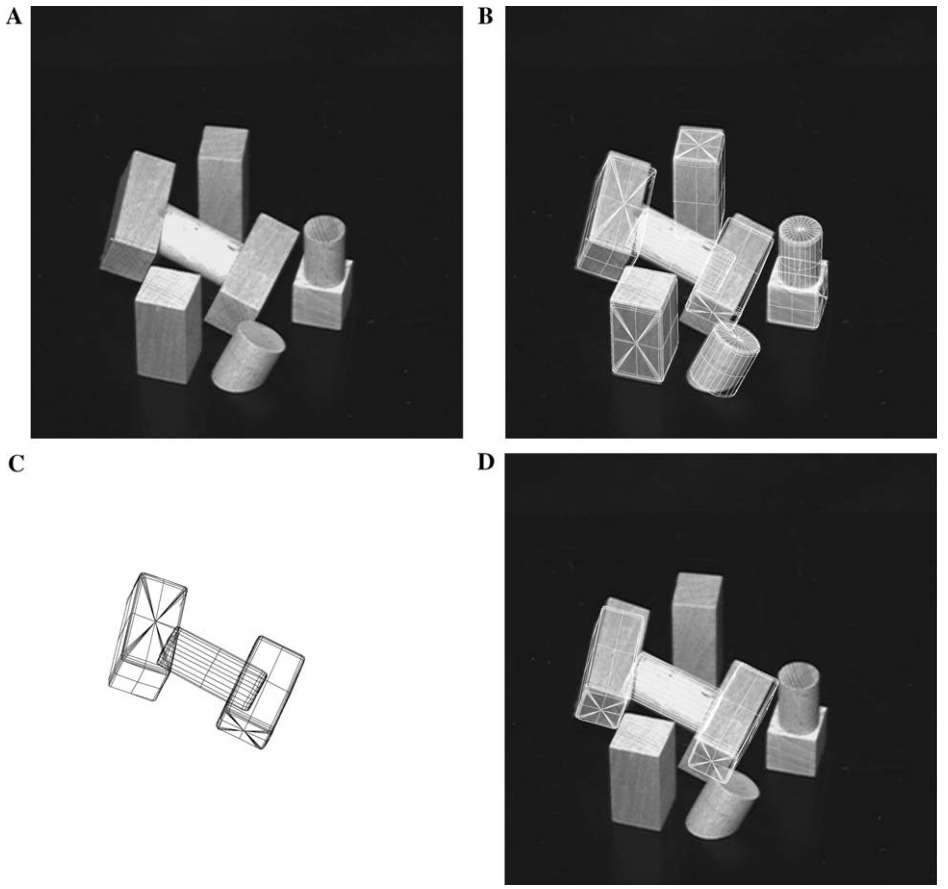

Fig. 9. Interpretation of a simple scene: (A) intensity image of a scene, (B) input range image with superimposed reconstructed superquadrics, (C) superquadrics selected for the hypothesis, (D) verification by refitting superquadrics of the model to corresponding segments in the range image.

found is a valid one, consisting of matches for all object parts, which are configured correctly.

As previously mentioned, the recognition system was tested using the figurine object on two types of scenes:

- scenes containing only one figurine in different configurations and
- complex scenes containing one or two figurines along with a large number of other parts.

With the first set of test images we wanted to test systematically the system's performance for isolated figurines. The figurine was configured into seven different poses and for each pose, range images from eight different viewpoints were captured, which makes a total of 56 images.

Fig. 10 shows one of the results, while Table 5 summarizes the recognition results. The object was detected in 39 cases. In 24 of those cases, the model computed from the best interpretation fitted the object very well. An interpretation included on the average 7.2 real matches. The object was not detected in 17 cases. In 9 of those cases, the reason for the failure was a singular object configuration as seen from that particular viewpoint. Due to occlusion, some parts, mainly the torso or the head, were not recovered properly, thus leading to a part configuration, which was later rejected when superquadric refitting was done. In the 8 other cases of failure the best interpretation found included less than five real matches, and was therefore rejected.

The system's performance was also tested on 20 different complex scenes. Complex scenes included several appearances of the figurine, as well as many unknown

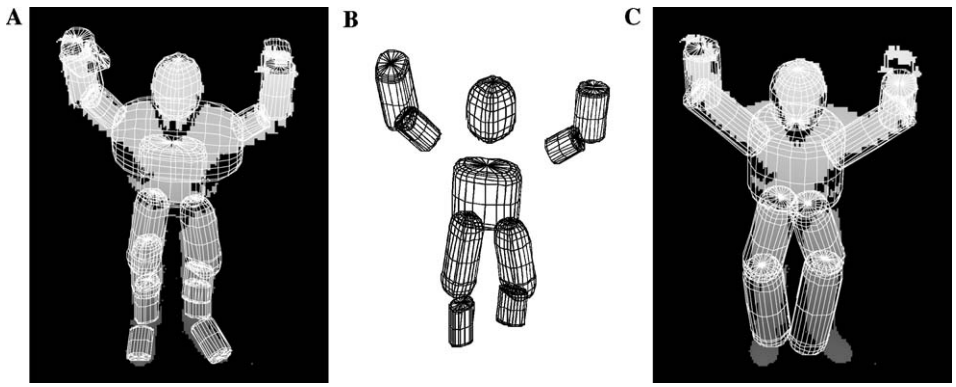

Fig. 10. Single figurine scene: (A) the input range image with superimposed reconstructed superquadrics, (B) superquadrics selected for the hypothesis, and (C) verification by refitting superquadrics of the model to their corresponding segments in the range image.

Table 5
Results of recognition on 56 scenes consisting of only one object

Total number of scenes: 56			
Object detected		Object not detected	
39		17	
Model fit		Cause	
Good	Poor	Occluded head or torso	Too few real matches
24	15	9	8

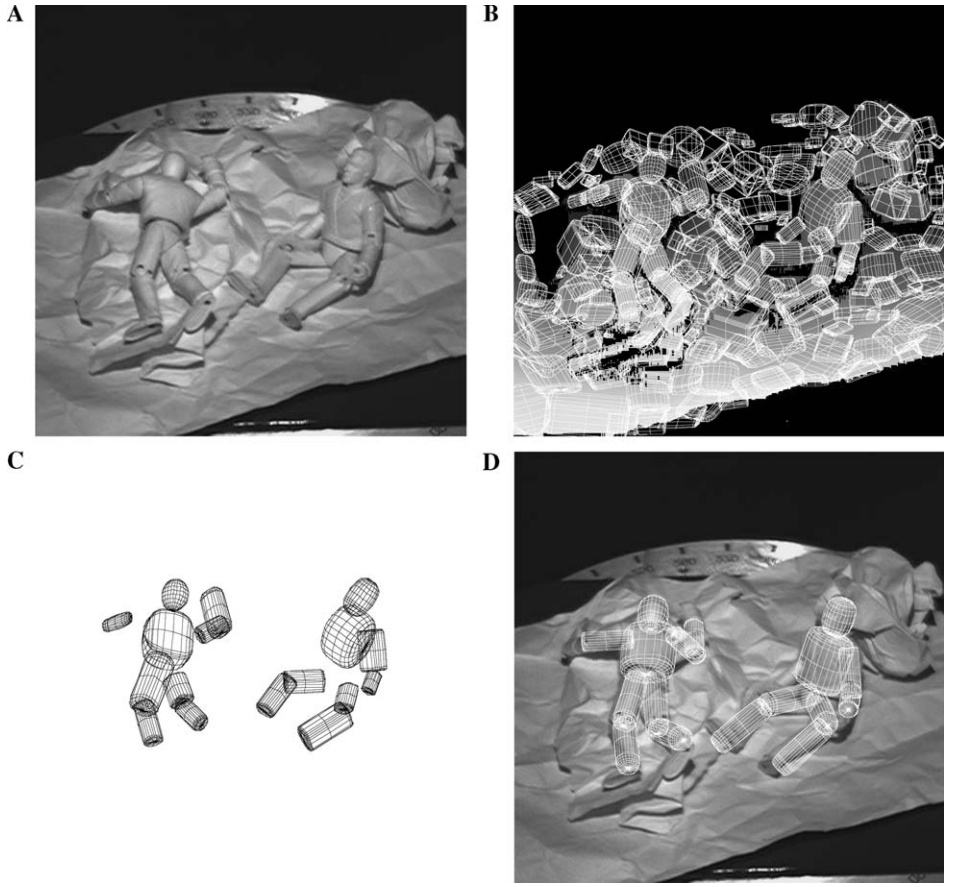

Fig. 11. Interpretation of a complex scene: (A) intensity image of a scene, (B) input range image with superimposed reconstructed superquadrics, (C) superquadrics selected for two hypotheses, and (D) verification by refitting superquadrics of the model to corresponding segments in the range image.

objects (Fig. 11). Nevertheless, there were no false positive recognitions of the human form, although there were many at least partially misleading part configurations. It is much harder to test a complex scene in a systematic fashion because of so many possible variables. One can observe that the reconstructions of the supporting surfaces in complex scenes were not appropriate, because such surfaces cannot be modelled well by superquadrics.

5. Conclusions

In this paper, we have investigated if superquadric-based shape decomposition can be used for recognition of articulated objects. The system is based on interpretation trees. We have shown, that despite very rough and sometimes unstable part

descriptions of natural shapes, superquadrics can be used in an object recognition scheme by introducing complex and sometimes model specific verification rules. The system can handle flexible articulated objects that cannot be perfectly modelled by superquadrics which is demonstrated by the recognition of the human figurine. Our approach should be useful for any kind of articulated objects with a clear part configuration.

The system could be improved in many ways. Best-first search could be implemented in order to inspect best interpretations (regarding part matches and/or scene part distances) first. Constraints could be added to limit the search based on object's size, so that the search would not include too distant scene parts. There is also a possibility for a parallel implementation of the whole system, including the segmentation and recovery stage which is currently the most time consuming part of the system.

If several models are used by the system, the obvious solution would be to use a separate interpretation tree for each object. But one can also come up with a single interpretation tree, which is especially useful in case that many objects share some parts. Nodes in such a tree would also carry the information about which object they can match. A node would expand to nodes representing all possible matches for all objects possibly matched by that node. After arriving to a leaf, the interpretation should be verified for all objects included in the leaf node. In case that none of the objects share the same part, the interpretation tree would consist of interpretation trees of all objects put side by side, all connected only to a single root node.

Acknowledgments

This work was supported by the Ministry of Education, Science and Sport of the Republic of Slovenia (Research program 1539-506). We would like to thank the anonymous reviewers for many valuable remarks that helped to improve the paper.

Appendix A

Algorithm 1. Interpretation tree search

// *Stack*—stack of nodes to be expanded
// *Interp*—list of consistent part matches that form an interpretation
// *MaxSize*—currently maximum interpretation size
// size(*Interp*)—no. of real part matches in *Interp*
// *root*—label for tree root
// consistent$(X, \mathcal{T})$—returns TRUE when X is a *root*, or X is a wildcard match, or X is $\mathcal{T}$ consistent real part match
// verify(*Interp*)—returns TRUE when *Interp* is sound
$Stack = [root], Interp = [], MaxSize = 0;$
WHILE (*Stack* not empty)
pop next match $X = (f_j, m_k)$ *from Stack*
determine the set of constraints $\mathcal{T}$ *for part* m_k

```
IF (consistent(X,T) AND max. possible interpret. size ⩾ MaxVel)
add X in Interp;
IF (leaf reached)
   IF (verify(Interp))
    save Interp;
    MaxSize = size(Interp)
   ENDIF
ELSE /* not a leaf, but still consistent */
   push (W, m_{k+1}) on Stack
   FOR i = 1...N
    IF (part f_i not in Interp)
       push (f_i, m_{k+1}) on Stack
    ENDIF
   ENDIF
ENDIF
ENDWHILE
```

References

[1] S.J. Dickinson, A.P. Pentland, A. Rosenfeld, From volumes to views: an approach to 3-D object recognition, CVGIP: Image Understand. 55 (2) (1992) 130–154.

[2] A. Jaklič, A. Leonardis, F. Solina, Segmentation and Recovery of Superquadrics, Kluwer Academic Publishers, Dordrecht, 2000.

[3] A.P. Pentland, Perceptual organization and the representation of natural form, Artif. Intell. 28 (1986) 293–331.

[4] N.S. Raja, A.K. Jain, Recognizing geons from superquadrics fitted to range data, Image Vision Comput. 10 (3) (1992) 179–190.

[5] F. Solina, R. Bajcsy, Recovery of parametric models from range images: the case for superquadrics with global deformations, IEEE Trans. Pattern Anal. Mach. Intell. 12 (1990) 131–147.

[6] A.H. Barr, Superquadrics and angle-preserving transformations, IEEE Comput. Graph. Appl. 1 (1981) 11–23.

[7] A. Leonardis, A. Jaklič, F. Solina, Superquadrics for segmentation and modeling range data, IEEE Trans. Pattern Recogn. Mach. Intell. 19 (11) (1997) 1289–1295.

[8] A. Leonardis, A. Gupta, R. Bajcsy, Segmentation of range images as the search for geometric parametric models, Int. J. Comput. Vision 14 (1995) 253–277.

[9] A. Leonardis, Image analysis using parametric models: model-recovery and model-selection, doctoral dissertation, University of Ljubljana, Faculty of Electrical Engineering and Computer Science, 1996.

[10] A. Jaklič, Construction of CAD Models from Range Images, doctoral dissertation. University of Ljubljana, Faculty of Electrical Engineering and Computer Science, 1997.

[11] F. Solina, A. Leonardis, A. Jaklič, B. Kverh, Reverse engineering by means of range image interpretation, in: P. Kopacek, D. Noe (Eds.), Intelligent Assembly and Disassembly, A Proceedings Volume from the IFAC Eorkshop, Bled, Slovenia, 21–23 May, 1998, Pergamon, Oxford, UK, 1998, pp. 153–158.

[12] F. Solina, A. Leonardis, Proper scale for modeling visual data, Image Vision Comput. 16 (2) (1998) 89–98.

[13] A. Chella, M. Frixione, S. Gaglio, Understanding dynamic scenes, Artif. Intell. 123 (2000) 89–132.

[14] I. Biederman, Human image understanding: recent research and a theory, Comput. Vision Graph. Image Process. 32 (1985) 29–73.

[15] R. Nevatia, T. Binford, Description and recognition of curved objects, Artif. Intell. 38 (1977) 77–98.

[16] R.C. Bolles, P. Horaud, 3DPO: a three-dimensional part orientation system, Int. J. Robotic Res. 5 (3) (1986) 3–26.
[17] W.Y. Kim, A.C. Kak, 3-D Object recognition using bipartite matching embedded in discrete relaxation, IEEE Trans. Pattern Anal. Mach. Intell. 13 (3) (1991) 224–251.
[18] W.E.L. Grimson, Object Recognition by Computer, MIT Press, Cambridge (MA), 1990.
[19] E. Trucco, A. Verri, Introductory Techniques for 3-D Computer Vision, Prentice Hall, NJ, 1998.
[20] V. Filova, F. Solina, J. Lenarčič, Automatic reconstruction of 3D human arm motion from a monocular image sequence, Mach. Vision Appl. 10 (1998) 223–231.
[21] L.H. Chen, Y.T. Liu, H.Y. Liao, Similarity measure for superquadrics, IEEE Proc. Vision Image Signal Process. 144 (4) (1997) 237–243.
[22] A. Jaklic, F. Solina, Moments of superellipsoids and their application to range image registration, IEEE Trans. Soc. Man Cybernetics–Part B: Cybernetics 33 (4) (2003) 648–657.
[23] N. Jojić, T.S. Huang, Computer vision and graphics techniques for modeling dressed humans, in: A. Leonardis, F. Solina, R. Bajcsy (Eds.), The Confluence of Computer Vision and Computer Graphics, Kluwer, Dordrecht, 2000, pp. 179–200.
[24] D. Skočaj, A. Leonardis, Acquiring range images of objects with non-uniform albedo using high-dynamic scale radiance maps, Proc. ICPR'00 (2000) 778–781.