Efficient Pose Estimation Using View-Based Object Representations

Gabriele Peters

Universität Dortmund, Informatik VII,
Otto-Hahn-Str. 16, D-44227 Dortmund, Germany,
peters@ls7.cs.uni-dortmund.de,
http://ls7-www.cs.uni-dortmund.de/~peters/

Abstract. We present an efficient method for estimating the pose of a three-dimensional object. Its implementation is embedded in a computer vision system which is motivated by and based on cognitive principles concerning the visual perception of three-dimensional objects. Viewpoint-invariant object recognition has been subject to controversial discussions for a long time. An important point of discussion is the nature of internal object representations. Behavioral studies with primates, which are summarized in this article, support the model of *view-based* object representations. We designed our computer vision system according to these findings and demonstrate that very precise estimations of the poses of real-world objects are possible even if only a few number of sample views of an object is available. The system can be used for a variety of applications.

1 Implications from Cognition

Each object in our environment can cause considerably different patterns of excitation in our retinae depending on the observed viewpoint of the object. Despite this we are able to perceive that the changing signals are produced by the same object. It is a function of our brain to provide this constant recognition from such inconstant input signals by establishing an internal representation of the object.

There are uncountable behavioral studies with primates that support the model of a view-based description of three-dimensional objects by our visual system. If a set of unfamiliar object views is presented to humans their response time and error rates during recognition increase with increasing angular distance between the learned (i.e., stored) and the unfamiliar view [1]. This angle effect declines if intermediate views are experienced and stored [2]. The performance is not linearly dependent on the shortest angular distance in three dimensions to the best-recognized view, but it correlates with an "image-plane feature-by-feature deformation distance" between the test view and the best-recognized view [3]. Thus, measurement of image-plane similarity to a few feature patterns seems to be an appropriate model for human three-dimensional object recognition.

J.L. Crowley et al. (Eds.): ICVS 2003, LNCS 2626, pp. 12–21, 2003.

Experiments with monkeys show that familiarization with a “limited number” of views of a novel object can provide viewpoint-independent recognition [4].

In a psychophysical experiment subjects were instructed to perform mental rotation, but they switched spontaneously to “landmark-based strategies”, which turned out to be more efficient [5] .

Numerous physiological studies also give evidence for a view-based processing of the brain during object recognition. Results of recordings of single neurons in the inferior temporal cortex (IT) of monkeys, which is known to be concerned with object recognition, resemble those obtained by the behavioral studies. Populations of IT neurons have been found which respond selectively to only some views of an object and their response declines as the object is rotated away from the preferred view [6].

The capabilities of technical solutions for three-dimensional object recognition still stay far behind the efficiency of biological systems. Summarizing, one can say that for biological systems object representations in form of single, but connected views seem to be sufficient for a huge variety of situations and perception tasks.

2 Description of the Vision System

In this section we introduce our approach of learning an object representation which takes these results about primate brain functions into account.

We automatically generate sparse representations for real-world objects, which satisfy the following conditions:

a1. They are constituted from *two-dimensional* views.

a2. They are *sparse*, i.e., they consist of *as few views as possible.*

a3. They are capable of *performing perception tasks*, especially pose estimation.

Our system consists of a *view representation builder* and an *object representation builder.* They are shown, together with their input and output data, in the diagram in figure 1, which depicts a one-directional flow of information.

Of course, feedback from higher levels of processing to lower ones would allow for, e.g., unsupervised system tuning or an improved segmentation, but this is not subject of this contribution. We start with the recording of a densely sampled set of views of the upper half of the viewing sphere of a test object. In the following we aim at choosing only such views for a representation which are representative for an area of viewpoints as large as possible.

2.1 View Representation Builder

Each of the recorded views is preprocessed by a *Gabor wavelet transform*, which is biologically inspired because Gabor wavelets approximate response patterns of neurons in the visual cortex of mammals [7,8]. A *segmentation* based on gray level values [9] follows. It separates the object from the background. This results

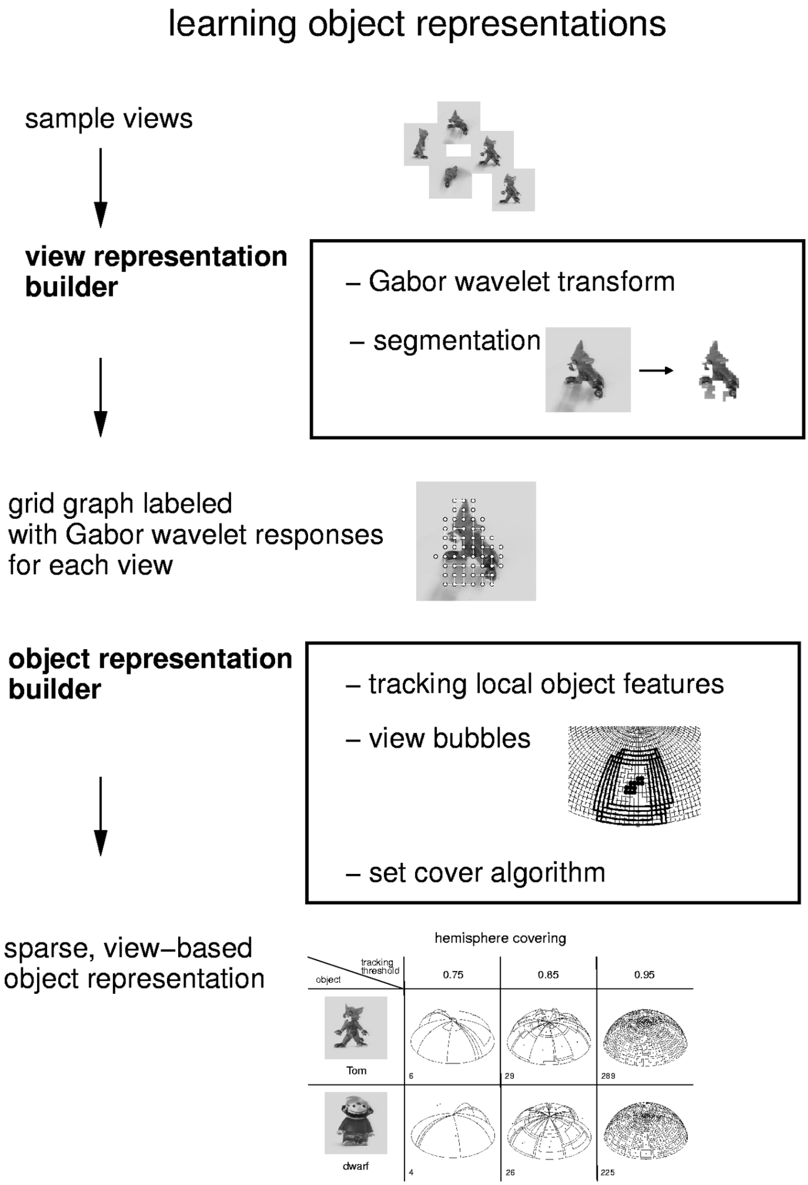
learning object representations
sample views
view representation builder
– Gabor wavelet transform
– segmentation
grid graph labeled with Gabor wavelet responses for each view
object representation builder
– tracking local object features
– view bubbles
– set cover algorithm
sparse, view–based object representation
hemisphere covering
tracking threshold
object
0.75
0.85
0.95
Tom
6
29
289
dwarf
4
26
225

Fig. 1. The system for learning sparse object representations consists of a view and an object representation builder. The resulting object representation consists of single but connected views. The numbers next to the resulting partitionings of the viewing hemisphere are the numbers of view bubbles which constitute the representation

in a representation of each view in form of a *grid graph labeled with Gabor wavelet responses*. The graph covers the object segment. Each vertex of such a graph is labeled with the responses of a set of Gabor wavelets, which describe the local surroundings of the vertex. Such a feature vector is called *jet*.

2.2 Object Representation Builder

To facilitate an advantageous selection of views for the object representation a surrounding area of similar views is determined for each view. This area is called *view bubble*. For a selected view it is defined as the largest possible surrounding area on the viewing hemisphere for which two conditions hold:

b1. The views constituting the view bubble are *similar* to the view in question.

b2. *Corresponding object points* are known or can be inferred for each view of the view bubble.

The similarity mentioned in **b1** is specified below. Condition **b2** is important for a reconstruction of novel views as, e.g., needed by our pose estimation algorithm. A view bubble may have an irregular shape. To simplify its determination we approximate it by a rectangle with the selected view in its center, which is determined in the following way.

The object representation builder starts by *tracking local object features*. Jets can be tracked from a selected view to neighboring views [10]. A similarity function $S(\mathcal{G}, \mathcal{G}')$ is defined between a selected view and a neighboring view, where $\mathcal{G}$ is the graph which represents the selected view and $\mathcal{G}'$ is a tracked graph which represents the neighboring view. Utilizing this similarity function we determine a *view bubble* for a selected view by tracking its graph $\mathcal{G}$ from view to view in both directions on the line of latitude until the similarity between the selected view and either the tested view to the west or to the east drops below a threshold τ, i.e., until either $S(\mathcal{G}, \mathcal{G}^w) < \tau$ or $S(\mathcal{G}, \mathcal{G}^e) < \tau$. The same procedure is performed for the neighboring views on the line of longitude, resulting in a rectangular area with the selected view in its center. The representation of a view bubble consists of the graphs of the center and four border views

$$\mathcal{B} := \langle \mathcal{G}, \mathcal{G}^w, \mathcal{G}^e, \mathcal{G}^s, \mathcal{G}^n \rangle \,, \tag{1}$$

with *w, e, s,* and *n* standing for *west, east, south,* and *north.* As this procedure is performed for each of the recorded views, it results in view bubbles overlapping on a large scale on the viewing hemisphere (see figures 1 and 2).

To meet the first condition **a1** of a sparse object representation we aim at choosing single views (in the form of labeled graphs) to constitute it. To meet the second condition **a2** the idea is to reduce the large number of overlapping view bubbles and to choose as few of them as possible which nevertheless cover the whole hemisphere. For the selection of the view bubbles we use the *greedy set cover algorithm* [11]. It provides a set of view bubbles which covers the whole viewing hemisphere. We define the *sparse, view-based object representation* by

$$\mathcal{R} := \{\langle \mathcal{G}_i, \mathcal{G}_i^w, \mathcal{G}_i^e, \mathcal{G}_i^s, \mathcal{G}_i^n \rangle\}_{i \in R} \tag{2}$$

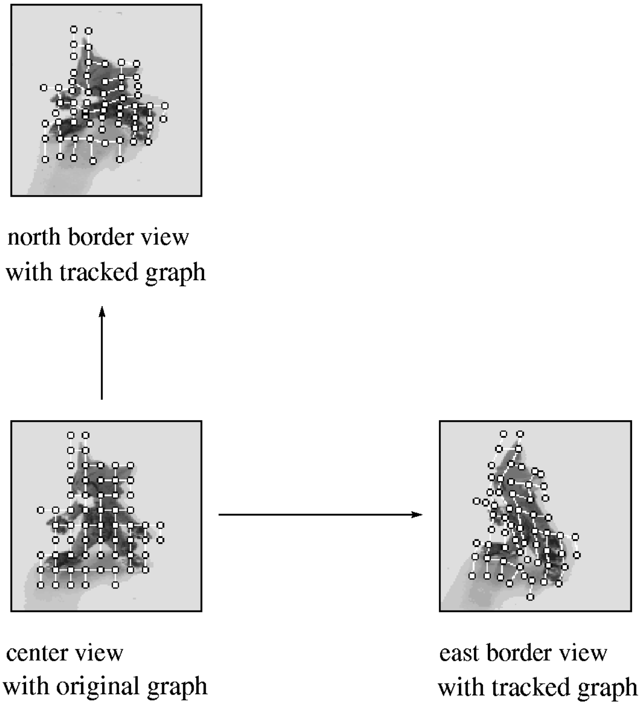

Fig. 2. This figure shows a graph of the center view of a view bubble tracked to its east and north border views

where R is a cover of the hemisphere. Neighboring views of the representation are “connected” by known corresponding object points (the correspondences between center and border views), which have been provided by the tracking procedure. Figure 1 shows different covers of the hemisphere for two test objects.

3 Pose Estimation

Given the sparse representation of the object in question and given a test view of the object, the aim is the determination of the object’s pose displayed in the test view, i.e., the assignment of the test view to its correct position on the viewing hemisphere. In this section a solution to this problem is proposed (subsection 3.1) and the results of simulations with a series of test views are reported (subsection 3.2) and discussed (subsection 3.3).

Many approaches to pose estimation have been proposed, starting from closed form solutions for not more than four noncollinear points [12,13,14] up to iterative non-linear optimization algorithms, which have to rely on a good initial guess to converge to a reasonable solution [15,16]. We propose a model based pose estimation algorithm. In a first step it determines the rough position of the given pose on the viewing hemisphere as initial guess. Then this estimate is refined in a second step. It requires the generation of *virtual views*, i.e., artificially generated images of unfamiliar views, which are not represented in the object representation. For this purpose we

(1) calculate linear combinations of corresponding vertex positions in the center and border graphs of view bubbles and

(2) interpolate the corresponding jets attached to these vertices.

The new positions and jets define a representing graph of the virtual view. From this graph the virtual view can be generated by reconstructing the information contained in Gabor wavelet responses [17]. To interpolate between jets we calculate the weighted sum of corresponding jets in the sample views. The weights are chosen according to the relative position of the unfamiliar view with respect to the sample views. Our method of deriving vertex positions in unfamiliar views follows Ullman and Basri's [18] purely two-dimensional approach of generating unfamiliar views by linear combination of sample views. Detailed formula are given in [19].

3.1 Methods

Let T be the test view, the pose of which should be estimated, and $\mathcal{G}_T$ be its representing graph, which is extracted from the original image of view T after the test view has been divided into object and background segments as described in section 2.1. This means that no a priori knowledge about the object is provided. A view is determined by its position on the viewing hemisphere.

Let $I_i, i \in R$, be the center images of the view bubbles the graphs $\mathcal{G}_i$ of the object representation $\mathcal{R}$ are extracted from. The *pose estimation algorithm* for estimating the pose of a single test view T proceeds in two steps:

1. Match $\mathcal{G}_T$ to each image $I_i, i \in R$, using a graph maching algorithm [20]. As a *rough estimate* of the object's pose choose that view bubble $\widehat{B}$ the center image I_i of which provides the largest similarity to $\mathcal{G}_T$.
2. Generate the representation $\widehat{\mathcal{G}}$ for each unfamiliar view which is included inside the area defined by $\widehat{B}$ by (1) a linear combination of corresponding vertex positions in the center and one border graph of $\widehat{B}$ and (2) an interpolation of the corresponding jets as described in section 3. (We choose the graph of that border view which lies closest to the unfamiliar view.) From each of the calculated graphs $\widehat{\mathcal{G}}$ reconstruct a corresponding virtual view $\widehat{V}$ using an algorithm which reconstructs the information contained in Gabor wavelet responses [17]. Accordingly, reconstruct a virtual test view $\widehat{V}_T$ from $\mathcal{G}_T$ (figure 3). Compare each of the virtual views $\widehat{V}$ with the virtual view $\widehat{V}_T$ using an error function $\epsilon(\widehat{V}, \widehat{V}_T)$ which performs a pixelwise comparison between $\widehat{V}_T$ and each $\widehat{V}$. The estimated pose $\widehat{T}$ of the test view T is the position on the viewing hemisphere of that virtual view $\widehat{V}$ which provides the smallest error ϵ.

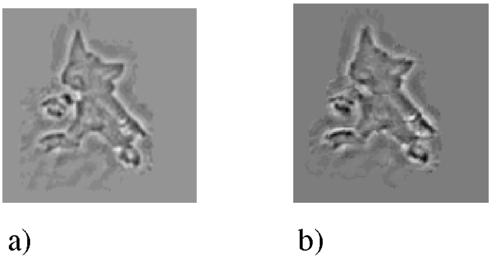

Fig. 3. **a)** Virtual view $\widehat{V}$ reconstructed from interpolated graph $\widehat{\mathcal{G}}$. **b)** Virtual test view $\widehat{V}_T$ reconstructed from its original graph $\mathcal{G}_T$

The *estimation error* between T and $\widehat{T}$ can be determined by the Euclidean distance: $\epsilon_{esti}(T, \widehat{T}) = d(T, \widehat{T})$.

3.2 Results

For the evaluation of the algorithm 30 test views have been chosen. The positions of them on the viewing hemisphere are displayed in figure 4. For two different toy objects and for three different partitionings of the viewing hemisphere, which have been derived by applying different tracking thresholds τ, the poses of these 30 test views have been estimated. The light gray squares indicate the views which are represented in the object representation $\mathcal{R}$, black dots mark the positions of the test images and the estimated positions are tagged by dark gray circles. The arrow points at the test images and their estimations which are displayed in figure 5.

The illustrations in figure 4 indicate that pose estimation becomes more precise with an increasing number of sample views in the object representation. This result has been expected and is confirmed by an inspection of the mean estimation errors taken over the 30 test views for each object and each partitioning of the hemisphere separately. They are summarized in table 1. With one exception for the “object” Tom the mean errors are decreasing with an increasing value of τ, i.e., with an increasing number of views in $\mathcal{R}$.

3.3 Discussion

The results of the pose estimation experiments are amazingly good. This is particularly obvious for the example displayed in figure 5, taking into account that the sparse representation of the object “Tom” contains only the representations of 30 views. This have been the test images for which the best result for $\tau = 0.75$ was obtained, but also for a reasonable partitioning of the viewing hemisphere ($\tau = 0.85$) the mean estimation errors are smaller than 5° for both objects, which can be regarded as a remarkable result, taking into account that humans are hardly able to recognize a difference of 5° between two object poses.

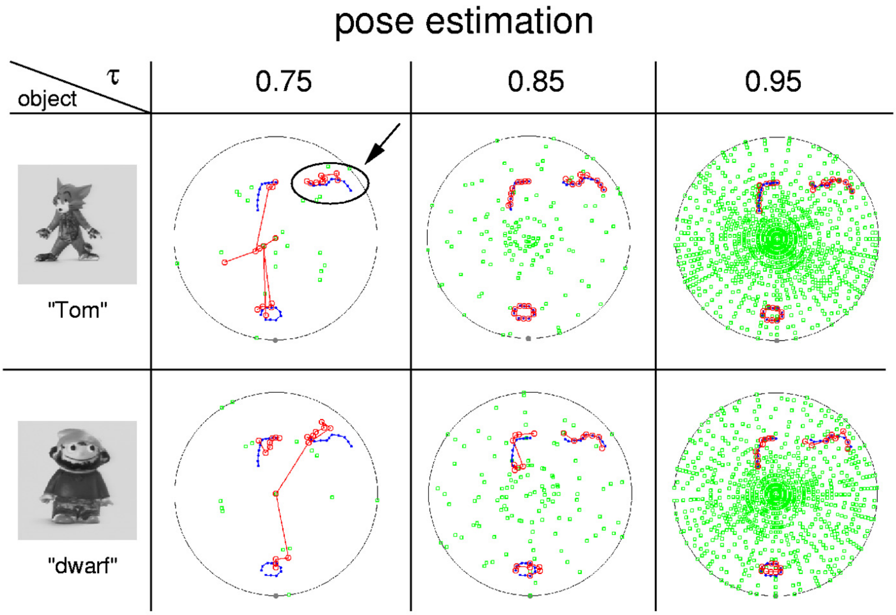

Fig. 4. Results of pose estimations for three different partitionings of the viewing hemisphere and two different objects are depicted. The tracking threshold τ influences the resulting number of views in the final representations. As for each view bubble of the final representation the graphs of the center and four border views are stored, the border views of neighboring view bubbles lie close together. This is obvious especially for $\tau = 0.75$

As experiments reported in [21] have shown, the method proposed in section 3.1 cannot be improved very much by a more elaborate determination of the initial guess, e.g., by testing more neighboring candidates.

4 Conclusion

We proposed a computer vision system based on cognitive principles which is able to estimate the pose of a three-dimensional object from an unobstructed view in an efficient manner. The pose estimation results support a good quality of our sparse object representation and allow the conclusion that a view-based approach to object perception with object representations that consist of only single views, which are connected, is suitable for performing perception tasks as it is advocated by brain researchers. Besides the biological relevance of our approach, there are a variety of possible applications, such as object recognition, view morphing, or data compression.

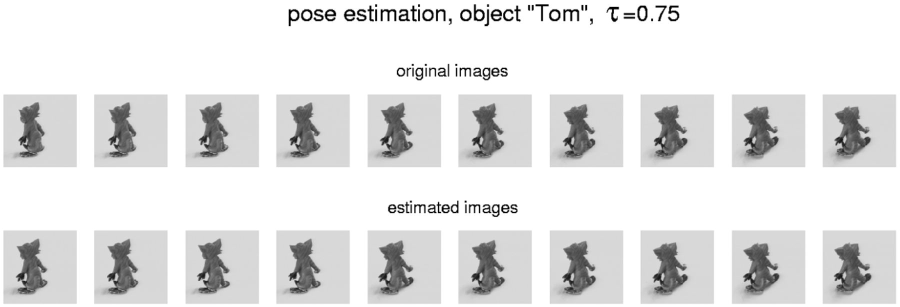

Fig. 5. This figure shows the test images and their estimations which are marked in figure 4. For this example the representation of the object "Tom" for $\tau = 0.75$ has been chosen. It consists of only 30 views. In the first row the true poses of the object, which should be estimated, are displayed. The second row shows the poses which have been estimated by treating each view of the sequence independently. The estimation error for this sequence averages 5.78°

Table 1. Mean pose estimation errors. For example, for object "Tom" and the partitioning of $\tau = 0.75$ the average estimation deviation of the estimated pose $\widehat{T}$ to the true pose T is 36.51°

mean pose estimation errors					
τ	0.75	0.8	0.85	0.9	0.95
object "Tom"	36.51°	3.63°	0.77°	3.35°	0.36°
object "dwarf"	20.54°	19.47°	4.2°	2.65°	1.71°

References

1. S. Edelman and H. H. Bülthoff. Orientation Dependence in the Recognition of Familiar and Novel Views of Three-Dimensional Objects. *Vision Research*, 32(12):2385–2400, 1992.
2. M. J. Tarr. *Orientation Dependence in Three-Dimensional Object Recognition.* Ph.D. Thesis, MIT, 1989.
3. F. Cutzu and S. Edelman. Canonical Views in Object Representation and Recognition. *Vision Research*, 34:3037–3056, 1994.
4. N. K. Logothetis, J. Pauls, H. H. Bülthoff, and Poggio T. View-Dependent Object Recognition by Monkeys. *Current Biology*, 4:401–414, 1994.
5. M. Wexler, S. M. Kosslyn, and A. Berthoz. Motor processes in mental rotation. *Cognition*, 68:77–94, 1998.
6. N. K. Logothetis, J. Pauls, and Poggio T. Shape Representation in the Inferior Temporal Cortex of Monkeys. *Current Biology*, 5(5):552–563, 1995.
7. D. C. Burr, M. C. Morrone, and D. Spinelli. Evidence for Edge and Bar Detectors in Human Vision. *Vision Research*, 29(4):419–431, 1989.

8. J. P. Jones and L. A. Palmer. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58(6):1233–1258, 1987.
9. C. Eckes and J. C. Vorbrüggen. Combining Data-Driven and Model-Based Cues for Segmentation of Video Sequences. In *Proc. WCNN96*, pages 868–875, 1996.
10. T. Maurer and C. von der Malsburg. Tracking and Learning Graphs and Pose on Image Sequences of Faces. In *Proc. Int. Conf. on Automatic Face- and Gesture-Recognition*, pages 176–181, 1996.
11. V. Chvatal. A Greedy Heuristic for the Set-Covering Problem. *Mathematics of Operations Research*, 4(3):233–235, 1979.
12. R. Horaud, B. Conio, O. Leboulleux, and B. Lacolle. An Analytic Solution for the Perspective 4-Point Problem. *Computer Vision, Graphics and Image Processing*, 47:33–44, 1989.
13. M. Dhome, M. Richetin, J. Lapreste, and G. Rives. Determination of the Attitude of 3-D Objects from a Single Perspective View. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(12):1265–1278, 1989.
14. R. M. Haralick, C. Lee, K. Ottenberg, and M. Nölle. Analysis and Solutions of the Three Point Perspective Pose Estimation Problem. In *Proc. of the IEEE Comp. Society Conf. on Computer Vision and Pattern Recognition*, pages 592–598, 1991.
15. D. G. Lowe. Three-Dimensional Object Recognition from Single Two-Dimensional Images. *Artificial Intelligence*, 31:355–395, 1987.
16. J. Yuan. A General Photogrammetric Method for Determining Object Position and Orientation. *IEEE Journal of Robotics and Automation*, 5(2):129–142, 1989.
17. M. Pötzsch. Die Behandlung der Wavelet-Transformation von Bildern in der Nähe von Objektkanten. Technical Report IRINI 94-04, Institut für Neuroinformatik, Ruhr-Universität Bochum, Germany, 1994.
18. S. Ullman and R. Basri. Recognition by Linear Combinations of Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):992–1006, 1991.
19. G. Peters and C. von der Malsburg. View Reconstruction by Linear Combination of Sample Views. In *Proc. BMVC 2001*, pages 223–232, 2001.
20. M. Lades, J. C. Vorbrüggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Würtz, and W. Konen. Distortion Invariant Object Recognition in the Dynamic Link Architecture. *IEEE Trans. Comp.*, 42:300–311, 1993.
21. G. Peters. *A View-Based Approach to Three-Dimensional Object Perception.* Ph.D. Thesis, Shaker Verlag, Aachen, Germany, 2002.