Learning the Topology of Views: 
From Images to Objects

Dissertation zur Erlangung des Grades 
eines Doktors der Naturwissenschaften 
der Fakultät für Physik und Astronomie 
an der Ruhr-Universität Bochum

vorgelegt von

Jan Wieghardt 
aus Heidelberg

Mai 2001
1. Gutachter: Prof. Dr. Christoph von der Malsburg
2. Gutachter: Dr. Marcel Kunze

Tag der Disputation: 19 Juli 2001
Acknowledgments

The here presented work is part of the ongoing research efforts by the groups of Prof. von der Malsburg at the Ruhr-University in Bochum and the University of Southern California in Los Angeles. I immensely enjoyed the superb working conditions and the creative atmosphere in both groups.

My special thanks go to Prof. von der Malsburg for creating this stimulating environment and for numerous open-ended inspiring discussions, which guided the course of this work.

I would also like to thank Dr. Rolf Würtz for his constant help and encouragement and Dr. Marcel Kunze for providing a physicist’s point of view and the second report.

My thanks also go to all my colleagues in Bochum and Los Angeles for their unconditional cooperation and a never ending exchange of ideas and resources.

Finally, I would like to thank Uta Schwalm and Michael Neef for keeping it all together.
Contents

1 Introduction 11
  1.1 Outline ................................................................. 14

2 Elastic Graph Matching 17
  2.1 Correspondences ....................................................... 17
  2.2 Local Image Description ............................................ 19
    2.2.1 Gabor Wavelet Transform .................................... 19
    2.2.2 Jets ............................................................... 20
    2.2.3 Similarity Functions ........................................... 21
  2.3 Graph Matching ....................................................... 23
    2.3.1 Graphs ........................................................... 24
    2.3.2 Matching ......................................................... 24
    2.3.3 Recognition ...................................................... 26
  2.4 Bunch Graph Matching ................................................ 26
    2.4.1 Features and Similarities ..................................... 26
    2.4.2 Recognition ...................................................... 27
  2.5 Discussion ............................................................ 27

3 Local Transformation Properties 29
  3.1 Transformation Properties ............................................ 29
    3.1.1 Generating Data ................................................. 30
    3.1.2 Statistical Analysis ........................................... 31
    3.1.3 Results .......................................................... 32
  3.2 Recognition with Object Knowledge .................................. 36
    3.2.1 Experiments ...................................................... 36
    3.2.2 Results .......................................................... 37
  3.3 Tracking with Object Knowledge ...................................... 39
    3.3.1 Tracking Single Points ....................................... 39
    3.3.2 Tracking Constrained Groups of Points ....................... 43
    3.3.3 Results .......................................................... 47
  3.4 Interpreting Data ...................................................... 50
## CONTENTS

3.4.1 Matching Principal Components to Simple Geometrical Deformations 50  
3.4.2 Principal Components and Rotation in Depth .......................... 52  
3.5 Conclusion ............................................................................. 54  

4 Applications 57  
4.1 Detailed Landmark Finding ....................................................... 57  
4.1.1 Hierarchical Processing ...................................................... 57  
4.1.2 Parameterization of Mapping Functions .............................. 59  
4.1.3 Results .............................................................................. 60  
4.2 Finger Tracking for Imitation Learning .................................... 62  
4.2.1 Overview ........................................................................... 62  
4.2.2 Phase-Based Tracking of Fingertips ................................. 63  
4.2.3 Discussion ........................................................................ 66  

5 Representation by Perceptual Manifolds 71  
5.1 Representational Spaces ........................................................... 71  
5.1.1 Physical Object Space ....................................................... 72  
5.1.2 Appearance Object Space ................................................ 72  
5.2 Retrieving the Topology of Views ......................................... 75  
5.2.1 Aspect Representations .................................................... 75  
5.2.2 Manifold Representation .................................................. 76  

6 Aspects 79  
6.1 View Representation by Labeled Graphs ................................ 79  
6.1.1 Segmentation .................................................................... 79  
6.1.2 Extracting and Matching Graphs ........................................ 80  
6.2 From Views to Aspects ............................................................ 81  
6.2.1 Local Linear Descriptions ................................................. 82  
6.2.2 Estimation of Topology Preservation ................................. 84  
6.2.3 Choosing a Threshold ...................................................... 85  
6.2.4 Estimating Local Dimensionality ..................................... 86  
6.3 Discussion ............................................................................. 87  

7 View Manifolds 89  
7.1 From Charts to Manifolds ....................................................... 89  
7.1.1 Approximating Aspect Distances ....................................... 89  
7.1.2 Global Parameterization .................................................. 91  
7.1.3 Aligning the Aspects ......................................................... 91  
7.1.4 Summary .......................................................................... 93  
7.1.5 Experiments ..................................................................... 94  
7.2 Learning to Estimate Pose Angles ........................................ 104  
7.2.1 Approximating Distances ................................................ 104
7.2.2 Interpolation on the Manifold ........................................ 105
7.2.3 Experiments ............................................................. 106

8 Discussion and Outlook .................................................. 109
  8.1 Biological Point of View ............................................. 110
  8.2 Continuity in Time .................................................... 111
  8.3 Active Handling of Objects .......................................... 112

A Principal Component Analysis ........................................ 115

B Metric Multidimensional Scaling ...................................... 119

C Zusammenfassung in deutscher Sprache .............................. 121
  C.1 Einleitung .................................................................... 121
  C.2 Elastische Graphenanpassung ....................................... 122
  C.3 Lokale Transformationseigenschaften ............................ 123
    C.3.1 Die Statistik von Transformationen ........................... 123
    C.3.2 Erkennung .......................................................... 124
    C.3.3 Punktverfolgung .................................................... 124
    C.3.4 Interpretation von Transformationen ........................ 125
  C.4 Anwendungen ............................................................ 125
    C.4.1 Detailliertes Finden von Landmarken ....................... 125
    C.4.2 Imitationslernen ................................................... 126
  C.5 Repräsentation durch Wahrnehmungsmannigfaltigkeiten .... 126
  C.6 Aspekte ..................................................................... 127
  C.7 Ansichtsmannigfaltigkeiten ......................................... 128
  C.8 Diskussion und Ausblick .............................................. 129

Bibliography ...................................................................... 131
List of Figures

2.1 Gabor Function .......................................................... 20
2.2 Similarity Potentials ...................................................... 21
2.3 Graph Matching .............................................................. 23

3.1 Example of automatically labeled faces ................................ 31
3.2 Principal Components of Face Graphs ................................... 33
3.3 Textured Principal Components of Face Graphs ......................... 34
3.4 Landmark Finding ............................................................ 38
3.5 Example of Background Suppression ..................................... 43
3.6 Tracking with and without Background Suppression .................... 44
3.7 Aperture Problem .............................................................. 46
3.8 Tracking with two Different Methods of Incorporating Constraints ... 48
3.9 Tracking of Faces .............................................................. 49
3.10 Selectivity of Principal Components .................................... 51
3.11 Principal Components under Rotation in Depth ......................... 53
3.12 Depth Related Principal Components ................................... 54

4.1 Facial Feature Finding ....................................................... 59
4.2 Stereo Manifold ............................................................... 66
4.3 Hand and Finger Detection ................................................ 68
4.4 Fingertip Tracking ............................................................ 68
4.5 Imitation .......................................................... 69

5.1 View Manifold ................................................................. 74

6.1 Single View Models .......................................................... 80
6.2 Examples of Aspect Representations ..................................... 84
6.3 Aspect Representation Parameters as Function of the Threshold $t$ ... 86
6.4 Estimating Dimensionality from Eigenvalues ........................... 87

7.1 Neighboring Aspects ......................................................... 92
7.2 Example Views ............................................................... 95
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.3</td>
<td>Evolution of Representations</td>
<td>96</td>
</tr>
<tr>
<td>7.4</td>
<td>Full View Topology (Pig and Box)</td>
<td>98</td>
</tr>
<tr>
<td>7.5</td>
<td>Full View Topology (Tom) in xy-plane</td>
<td>99</td>
</tr>
<tr>
<td>7.6</td>
<td>Full View Topology (Tom) in xz-plane</td>
<td>100</td>
</tr>
<tr>
<td>7.7</td>
<td>Full View Topology (Ink Blotter)</td>
<td>101</td>
</tr>
<tr>
<td>7.8</td>
<td>Full View Topology (Animal)</td>
<td>102</td>
</tr>
<tr>
<td>7.9</td>
<td>Full View Topology (Dwarf)</td>
<td>103</td>
</tr>
<tr>
<td>7.10</td>
<td>Interpolation Error</td>
<td>107</td>
</tr>
<tr>
<td>7.11</td>
<td>Error Distribution</td>
<td>108</td>
</tr>
<tr>
<td>A.1</td>
<td>Geometric Interpretation of Principal Component Analysis</td>
<td>116</td>
</tr>
<tr>
<td>A.2</td>
<td>Scree-Test</td>
<td>117</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Vision is the prime source of information about the environment for most mammals. Making this source of information available to machines has been a major field of research for the past decades. But the ease with which humans perceive their environment while recognizing objects, navigating, or interacting in any other way with their surroundings, belies the underlying difficulties. The following example, as cited in (Wallis and Bulthoff, 1999), illustrates the amount of abstraction involved in our perception of the world.

Quite recently [S.B.] had been struck by how objects change their shape when he walked around them. He would look at a lamp post, walk around it, stand studying it from a different aspect, and wonder why it looked different and yet the same. Richard Gregory and Jean Wallace (Gregory and Wallace, 1963)

The above described subject was a man who was able to see after fifty years of blindness thanks to an operation. The source of his bewilderment is rooted in some very fundamental problems regarding vision. The goal of achieving a stable and useful interpretation of the environment is obstructed by the fact that the projection of the world perceived by the visual senses is subject to very rapid change. One possible source of change are changes in viewpoint or, more generally put, changes in the geometrical observer-object relation, as experienced by S.B.. These result in severe changes of the projected images on the retina. To perceive a stable environment a machine or a brain must make sense out of all those different views projected onto the retina. To this end an internal representation of an object must be established. The nature of a suitable internal object representation is a much disputed subject (see (Edelman, 1997; Wallis and Bulthoff, 1999) for comprehensive reviews), but some basic requirements must certainly be fulfilled.

Firstly, the internal representation must be reliably linked to the visual input, i.e. known visual appearances or views of an object must be recognized as such. As one cannot assume that every possible view of an object is stored by or even known to the system, the procedure of object or rather view recognition requires some sort of generalization beyond the known views (model views). Existing computer vision algorithms which are concerned
with object recognition vary considerably in terms of the amount of generalization deemed necessary, the sources of variation considered, and the object features assumed suitable for representation. *Elastic Graph Matching* (Lades et al., 1993), as discussed in chapter 2, uses, e.g., local image features which are fairly insensitive to changes in illumination and scale, connected to build a graph, which encodes the neighborhood relations between the local features. By using fairly robust image features and by allowing their relations to vary moderately between two different views a generalization over slight variations is achieved. A more ambitious approach is taken by the *SEEMORE* system (Mel, 1997). Here a large number of local image features, partly similar to the ones employed in *elastic graph matching*, are used to represent given object appearance. But any relation between the local features is discarded in order to achieve generalization over large changes in viewpoint. In comparison to elastic graph matching this unqualified removal of all neighborhood relation creates an enormous amount of ambiguity. It is thus not clear whether generalization may come at the price of decreased discriminatory powers. In other cases (Turk and Pentland, 1991; Murase and Nayar, 1995) views of objects are not represented as an ensemble of local image features, but a hierarchy of linear filters derived via *principal component analysis* (see appendix A) from a set of given model views. Generalization is in these cases achieved by judging whether or not a deviation from some prototypical view can be reconstructed well from the filter output, such that it can be assumed to result from a typical deformation, or not.

The second requirement for object representations stems from the underlying need to interact with objects as well as to predict invisible views. The necessity to extrapolate beyond the currently visible by exploiting knowledge about the perceived object, leads to the notion of the so called *second order isomorphism* (Shepard, 1968; Shepard and Chipman, 1971; Edelman and Duvdevani-Bar, 1997). *First order isomorphism* of representation is the idea that the representation of something is similar to the object represented. *Second order isomorphism* on the other hand requires that the instances of representation exhibit the same relations as the images of the object being represented. For example from the fact that a cat shares more properties with a dog than with a book, one could assume that the representations of a dog and a cat are also more similar to each other than the representations of a cat and a book. In the same spirit one must assume that different views of one object are represented in a way that captures the distance of those views in terms of physical variables, such as, e.g., rotation angle. Putting it differently, a topology or even metric on the different views must be part of any object representation, because it allows to judge the closeness to situations experienced before, such that meaningful inter- and extrapolation becomes feasible. Experiments reported in (Wang et al., 1996) suggest that these ideas are also of biological relevance. They observed activity distributions in the *inferotemporal cortex* of a monkey, an area of the brain linked to object recognition, while the monkey observed heads rotating in depth. Their results show, that the amount of spatial shift of activity in the cortex is closely related with the amount of rotation of the heads, thus the brain seems to create a topological mapping of the views of heads
in terms of corresponding rotation angles. In terms of computer vision this second order isomorphism can be implemented in a straightforward fashion by labeling the model views with the relevant physical parameters obtained from a source outside the visual system. In (Krüger et al., 1996; Becker et al., 1999) a rough pose estimation was developed via a nearest neighbor search in view space, using the same mode of comparison employed in the object recognition part of the system. Such a nearest neighbor search can of course be easily refined if the recognition mode supplies a vector space structure on the view space. This can then be exploited to interpolate between the labeled model views (e.g. (Murase and Nayar, 1995; Lanitis et al., 1997)). Another straightforward implementation is that of an explicit 3-D model. Such a model could easily be parameterized in terms of physical parameters via the well understood properties of the geometrical rotation groups. But, even putting aside the overwhelming evidence from computer vision and psychophysics that such a representation plays no role in biological systems (Bülthoff and Edelman, 1992; Edelman, 1997; Wallis and Bülthoff, 1999), it would be hard to see how such an approach could satisfy the third major requirement of object representation.

Any representation of the visual appearance of a given object must be derivable from the information available to the system, i.e. it must be learnable with reasonable effort. Frequently the learning of object representations is just seen as a search for invariances, i.e. the search for feature combinations distinctive for one object or object class, but invariant under all kinds of transformations of the object’s appearance (see for example (Fukushima, 1980; Poggio and Edelman, 1990; Rolls, 1995; Wallis and Baddeley, 1997; Riesenhuber and Poggio, 1999; Amit, 2000)). In other cases the possible transformations of an object are fixed and not subject to learning. The transformations are either taken from a priori assumptions or from typical examples. Learning occurs here as a generalization from one instance of a class of objects to another by subjecting the visual features of a new object to those transformations which were derived beforehand. This approach usually leads to one shot learning, where single views of new objects are sufficient for recognition (see for example (Lades et al., 1993; Beymer and Poggio, 1996; Mel, 1997; Lanitis et al., 1997; Vetter and Poggio, 1997)). This type of approach is usually only applicable for so called nice object classes (Logothetis et al., 1994), i.e. classes of objects which share common transformation properties. An exception is (Mel, 1997), which allows for a wide range of transformations at the price of overgeneralization as discussed above. To tackle the learning of invariances as well as the transformation properties of novel objects another approach is frequently employed. Here the model views of an object are labeled with additional information which allows to establish a sensible metric on the views. Provided with this information unknown views can be derived and interpreted via various inter- and extrapolation techniques. Examples of this kind of supervised learning can be found in (Murase and Nayar, 1995; Walter, 1996; Kefalea, 1998; Kröse et al., 1999; Peters, 1999; Okada et al., 2000; Tenenbaum and Freeman, 2000). The information with which the views are labeled depends of course on the specific application scenario, but usually the views are associated with parameters of some abstract physical state.
space, such as rotation angles or coordinates of fiducial points in the three-dimensional Euclidean space. Such spaces are deemed to provide a good representation for most applications. An unsupervised method of introducing a metric on a set of views was outlined in (Tenenbaum, 1998; von der Malsburg et al., 1999; Wieghardt and von der Malsburg, 2000; Tenenbaum et al., 2000). These can be used if the precise properties of the parameterization are of no concern. Here the local neighborhood structure of views is exploited to estimate the degrees of freedom of a given object and to establish a topology.

1.1 Outline

This thesis mainly deals with the above discussed concept of object representation. We will try to go all the way from single views over the aspects of an object, i.e. sets of similar views with known transformation relations, to complete representation of the viewsphere, i.e. the set of all possible views of an object. The main focus is on the internal representation of transformations, especially those caused by changes in observer-object relations, as well as the learning from natural data. The issue of object recognition and related subjects such as the tracking of objects are also touched. More advanced topics related to object classification, recognition by parts, and generalization over large classes of objects are beyond the scope of this work.

In chapter 2 elastic graph matching (Lades et al., 1993) and bunch graph matching (Wiskott et al., 1997) are introduced. These algorithms have shown to provide very good solutions to the problem of recognizing views under slight variations as well as generalization over nice classes from few examples. They constitute the backbone of our object representation. The introduction will especially focus on the question, how elastic graph matching achieves the necessary generalization.

Chapter 3 describes how an object representation for faces can be bootstrapped from very little a-priori knowledge, if some restrictions on the amount of variation between different views are imposed. To this end the typical transformations of faces are extracted and incorporated into a linear model in a way very similar to (Lanitis et al., 1997). But the described approach goes beyond (Lanitis et al., 1997), as the correspondences (see chapter 2) are resolved automatically and the training database was not explicitly chosen to reflect certain variations. Moreover, it is discussed how the acquired representation of faces aids the recognition process (section 3.2) and the retrieval of correspondences in a tracking scenario (section 3.3). It is further discussed to what extent the model parameters reflect actual properties of the real world or in other words, whether a second order isomorphism can be established based on the derived linear model.

In chapter 4 two example applications of the techniques developed in chapter 3 are presented. The first application deals with the retrieval of fiducial points in high resolution face images (section 4.1). The linear representations of typical transformations as described in chapter 3 are employed to solve the problem with high accuracy. The second
1.1. OUTLINE

application (section 4.2) is concerned with the tracking of finger tips in stereo images with the goal of enabling imitation learning of grasping movements by an anthropomorphic robot (Triesch et al., 1999). The focus will be on the application of linear techniques in the presence of a non-linear constraining manifold.

Chapters 5 to 7 addresses the problem of creating a representation of a novel object from scratch, without any constraints in terms of the admissible observer-object relations. To this end some general remarks are made in chapter 5 concerning the nature of desirable representational spaces and what can be derived from visual inspection alone. The approach presented afterwards in chapter 6 is also based on linear approximation, thus preserving the advantages in connection with object recognition and tracking demonstrated in chapter 3.

Chapter 7 then provides the means to integrate the local linear models into one coherent object representation, rendering a complete understanding of the transformation properties of the object. The representation is established only on the basis of visually accessible information, the need for an external ground truth parameterization of views does not arise, distinguishing it from many other approaches (see above). Because the approach is explicitly designed to serve as a learnable object representation rather than a mere tool for dimensionality reduction, it also goes beyond the algorithms proposed in (Tenenbaum et al., 2000), with whom it shares some features. It will also be shown that, although the retrieved metric on the views is solely based on visual appearance, it aids significantly in associating views with parameters of the physical world and is thus a suitable basis for interaction with the represented objects. To this end a primitive learning algorithm is devised, which allows to associate visual appearance with viewing angles.
Chapter 2

Elastic Graph Matching

2.1 Correspondences

Two projections of the same object, as seen in two images, may differ strongly, depending, among other things, on the point of view, the current illumination, or occlusion by other objects. The task to identify those points in the two projections which stem from the same points on the object is termed the correspondence problem. A full correspondence can be defined as follows:

**Correspondence:** The mapping which assigns a point in one image of an object to its unique corresponding point in another. Those pairs of points are deemed corresponding which are projections of the same point on the object.

The correspondence problem occurs in a number of tasks related to vision, such as stereo vision, motion estimation, and object recognition, because vital information about the physical world is contained in the correspondence. In stereo vision point correspondences need to be established between two images taken at the same moment in time of the identical physical scene from two different points of view in order to determine relative differences in depth from the disparities of those pairs of points. In motion estimation tasks correspondences can be used to extract motion from two images received by the same observer at different instances in time. In object recognition correspondences between a stored model and new input must be established to allow identification by comparing local image features. As much as a solution to the correspondence problem is desirable as difficult it has turned out to obtain one. The FERET face recognition test performed by the Army Research Laboratory in 1998 (Phillips et al., 1999) can serve as an account of this fact. The aim of this test was to evaluate state of the art automatic face recognition systems. To this end large galleries of face images were recorded under different conditions. The performance of the different participating systems was then measured in terms of correct recognitions. The test was separated into two distinct parts, one with only the various images given and another one where additional information on
the eye positions of the shown faces was supplied. 10 out of 12 algorithms required the solution of the correspondence problem to be given a priori in form of the eye coordinates. One of the two fully automatic systems performed significantly worse without the additional information about the location of the eyes. The only system which performed satisfactorily even when no solution to the correspondence problem was provided, was based on the elastic graph matching approach (Lades et al., 1993; Würtz, 1995; Wiskott, 1995; Wiskott et al., 1997), which also serves as a basis to the work presented here.

In order to establish the required point correspondences a point description is needed which allows to retrieve a given point from an image. Characterizing a point in terms of its gray value is clearly not sufficient. In general each gray value is shared by a number of points in an image, and the description is thus highly ambiguous. The gray value is also sensitive to changes in illumination and is thus not a reliable feature. It is therefore inevitable to describe a point in relation to other points. But this introduces new problems, because these relations between gray values may be subject to change, if for example the object shown in an image is rotated. To deal with these changes in relation all the possible different relations can be included in the point description or alternatively the transformation properties of these relations can be included. If the ensemble of points, whose relations are considered, is well chosen these relation transformations are subject to stringent constraints. If for example relations between points belonging to a single rigid object are considered, only those relation transformations are feasible which are consistent with the transformation properties of rigid bodies. If on the other hand relations between points on different objects are considered nothing constrains the possible relation transformations, thus providing no advantage over a single-point description.

It is therefore desirable to increase the number of coded relations as long as no newly introduced relation transformations offset the gain in ambiguity reduction. Following this argument correspondence must be established for a whole ensemble of points, such as all points on a single object. This way the question of, “which point in one image corresponds to which point in another”, turns to, “which one of a given set of mappings $F$ is the actual correspondence”.

The actual correspondence mapping $\tilde{f}_c(\tilde{x})$ then contains the desired information on the current state of the physical world, the mapping set $F$ contains everything to be known about the problem a priori, that is before the current state is evaluated. For example the fact that finding stereo correspondences is a problem with only one local degree of freedom is reflected by a suitable choice of an $F$ containing only those functions $\tilde{f}_i(\tilde{x})$ which map a point somewhere on its epipolar line (Marr, 1982). This example also shows that a single parameter, here the depth of a point, can be used to parameterize the mappings. This in turn can be exploited to introduce a structure onto the mapping set $F$, thus allowing for systematic searches upon $F$. A major part of this thesis will be devoted to the question on how to choose a suitable set $F$, how to impose structure upon it, and in what way this structure reflects properties of the physical world.
2.2. LOCAL IMAGE DESCRIPTION

In the following sections elastic graph matching (Lades et al., 1993; Würzt, 1995; Wiskott, 1995; Wiskott et al., 1997) is introduced, which has proven, for example in the above mentioned FERET test, to solve the correspondence problem well enough for object recognition tasks.

2.2 Local Image Description

The goal of a local image description must be to reduce the ambiguity in the description of a single point, in order to facilitate the retrieval of correspondences. Moreover, such a representation should exhibit some degree of invariance under local transformations to allow for generalization over different projections of the same feature. This and more is provided by features based upon a Gabor wavelet transform.

2.2.1 Gabor Wavelet Transform

A Gabor transform of an image $I$ at a point $\vec{x}$ is given by its convolution with a complex Gabor function $\psi_{k}$, which is basically a plane wave restricted by a Gaussian envelope (see figure 2.1 for an example).

$$J_{k}(\vec{x}) = \int I(\vec{x}') \psi_{k}(\vec{x} - \vec{x}') d^{2}x'$$

$$\psi_{k}(\vec{x}) = \frac{k^2}{\sigma^2} \exp \left(-\frac{k^2 x^2}{2\sigma^2}\right) \left[\exp \left(i\vec{k}^{T} \vec{x}\right) - \exp \left(-\frac{\sigma^2}{2}\right)\right]$$

By rotating and scaling the wave vector $\vec{k}$ a whole family of Gabor functions can be derived, all parameterized in terms of their orientation $\Phi$ and frequency $k$.

$$\vec{k} = k \begin{pmatrix} \cos \Phi \\ \sin \Phi \end{pmatrix}$$

The Gabor functions and their associated transforms have some unique properties, which make them well suited for coding local image features.

- They are mean free and therefore insensitive to homogeneous changes in illumination.

- They are normalized to compensate for the effect of decaying energy content at higher frequencies in natural images, which was shown by David Field (Field, 1987) to be approximately proportional to $\frac{1}{k^2}$. The normalization factor $\frac{k^2}{\sigma^2}$ thus yields similar response distributions on all frequency levels.

- The kernels are localized in frequency as well as space domain. In the limit of very large $\sigma$ they are even maximally localized in frequency and space domain
Figure 2.1: **Gabor Function:** Gabor functions have the shape of a plane wave restricted by a Gaussian envelope. They are ideally suited for image representation, because of their properties regarding information theory (Olshausen and Field, 1996; Bell and Sejnowski, 1997) and biological relevance (Jones and Palmer, 1987). Shown are the real and imaginary part of a Gabor function in arbitrary units.

(Würtz, 1995). Thus they characterize a point well, while considering only a small neighborhood.

- Gabor shaped filters were found to be optimal in coding images of natural scenes in terms of information theoretical properties, such as sparseness and minimal mutual information (Olshausen and Field, 1996; Bell and Sejnowski, 1997). A certain response distribution over a set of Gabor filters is therefore a very characteristic signal. Thus these filters fulfill the requirement of ambiguity reduction.

- Gabor filters have also been shown to approximate the response properties of cortical neurons in primary visual cortex of mammals (Jones and Palmer, 1987) and seem to be of high biological relevance for primary vision.

### 2.2.2 Jets

The complex responses of a set of Gabor filters of different orientations and scales at a given location in an image constitute a so called jet (Lades et al., 1993). These jets are vectors of complex numbers $\mathbf{j}^T = \left( a_k e^{i\phi_k}, \ldots, a_n e^{i\phi_n} \right)$, which are characterized by their associated set of filters (see equation 2.1). The finite set of filters is chosen such that the direction space is sampled homogeneously

$$\Phi_\nu = \frac{\nu \pi}{N} \text{ with } \nu \in \{0, \ldots, N - 1\}$$ (2.4)
2.2. LOCAL IMAGE DESCRIPTION

Figure 2.2: **Similarity Potentials:** Displayed are the potentials of different measures of similarities for a single jet taken from the center of some arbitrary image showing a frontal face. The similarity measures used were the absolute value based similarity (\(\text{abs}\)) (Lades et al., 1993), the absolute and phase based similarity (\(\text{phase}\)) (Lades et al., 1993), and the absolute and compensated phase based similarity (\(\text{disp}\)) (Würtz, 1995; Wiskott, 1995). In the translation case (a) a jet was extracted and compared to jets derived at positions along a line of increasing distance from the original position. The sensitivity to scaling was tested by increasing and decreasing the image from its original size 128 \(\times\) 128 pixels and comparing the original jet to the jets at corresponding positions in the scaled images. Rotation (c) was tested by rotating the image around the point at which the original jet was extracted and comparing the original jet to jets extracted at the same pixel position in the rotated images.

and the frequencies are sampled geometrically

\[ k_\mu = \frac{k_{\text{max}}}{k_{\text{step}}} \quad \text{with} \quad \mu \in \{0, \ldots, M - 1\} \]  

(2.5)

The remaining parameters are chosen according to (Wiskott, 1995).

\[ k_{\text{step}} = \sqrt{2} \quad k_{\text{max}} = \frac{\pi}{2} \quad N = 8 \quad M = 5 \quad \sigma = 2\pi \]  

(2.6)

These parameters realize a complete coverage of the frequency domain with little overlap between the filters for images of 128 \(\times\) 128 pixels size. For larger images more low frequency components must be added, i.e. \(M\) must be increased.

2.2.3 Similarity Functions

To assess whether two points from two different images actually correspond to each other, a measure of similarity between the local features is needed. Such a similarity measure should fulfill certain requirements. It should be invariant under those changes in the
image, which are irrelevant to the correspondence problem, such as for example illumination. Local transformation, such as rotation and scaling, should yield smooth similarity potentials to aid the search for correspondences in the transformation space. And lastly the similarity function should not introduce new unsolicited ambiguities. To this end three different similarity functions are introduced, which fulfill different requirements to different degrees.

1. **Similarity based on absolute values only** (abs):

   \[ s_{\text{abs}} = \frac{\sum_k a_k a_k'}{\sqrt{\sum_k a_k^2 \sum_k a_k'^2}} \]  

   (2.7)

   Because most of the variations in a jet are captured by the phase of the Gabor responses, especially in the case of translation, \( s_{\text{abs}} \) yields smooth similarity potentials, with fairly wide maxima.

2. **Similarity based on phases and absolute values** (phase):

   \[ s_{\text{phase}} = \frac{\sum_k a_k a_k' \cos(\phi_k - \phi'_k)}{\sqrt{\sum_k a_k^2 \sum_k a_k'^2}} \]  

   (2.8)

   \( s_{\text{phase}} \) considers also the phase differences between two jets. The similarity generated by \( s_{\text{phase}} \) is thus very sensitive towards small image transformations, but can be used effectively to distinguish between very similar correspondences.

3. **Similarity based on compensated phases and absolute values** (disp):

   \[ \hat{d} = \arg \max_{\tilde{d}} \left\{ \frac{\sum_k a_k a_k' \cos(\phi_k - \phi'_k - \tilde{k}^T \tilde{d})}{\sqrt{\sum_k a_k^2 \sum_k a_k'^2}} \right\} \]  

   (2.9)

   \[ s_{\text{disp}} = \frac{\sum_k a_k a_k' \cos(\phi_k - \phi'_k - \tilde{k}^T \tilde{d})}{\sqrt{\sum_k a_k^2 \sum_k a_k'^2}} \]  

   (2.10)

   \( s_{\text{disp}} \) is derived from \( s_{\text{phase}} \), but two additional degrees of freedom are introduced, which allow to compensate for the change in the phases due to translation explicitly. The observation is used that the phases behave nearly linearly as a function of translation. This effect can be exploited to estimate the correct disparity \( \hat{d} \) between the two jets. So \( s_{\text{disp}} \) yields not only a similarity but also a disparity estimation. The actual evaluation of \( s_{\text{disp}} \) is discussed in section 3.3.1 in the context of tracking, but it should be mentioned that the similarity as well as the disparity are estimated using a linearization of equation 2.9 and equation 2.10, such that the close relation between equation 2.10 and equation 2.8 only holds for small disparities. Moreover, the estimation is limited to those disparities which are smaller than half the wavelength of the lowest frequency Gabor filter.
2.3. GRAPH MATCHING

Given a local image description associated with a similarity measure, as for example the one introduced above, *elastic graph matching* as described in (Lades et al., 1993; Würtz, 1995; Wiskott, 1995; Wiskott et al., 1997) provides the means to establish the correspondence between a model image and an input image. The general notion is, that taking into account the ambiguities inherent in a local image description the correspondence must be established as a whole and this can be achieved by generating a model description based

In order to achieve invariance under global changes in contrast, all similarity functions are normalized by the absolute length of each jet. The behavior of these similarity functions under local deformations of the image is shown in figure 2.2 for one example jet and three typical image transformations – scaling, rotation, and translation.

Figure 2.3: **Graph Matching**: Shown is a schematic overview of the elastic graph matching algorithm as described in (Lades et al., 1993; Würtz, 1995; Wiskott, 1995; Wiskott et al., 1997). After extracting local image features, so called jets, from a given image, different mappings, chosen from basic sets of mappings between a model graph and the image representation (see gray box) are evaluated according to local feature similarities. The mapping, to which the highest total similarity is assigned is assumed to reflect the actual correspondence between model graph and image best.

2.3 Graph Matching
on a set of local features and a hierarchy of mapping functions, which in turn can be evaluated in terms of their implied local feature correspondences.

2.3.1 Graphs

To represent an object in an image in terms of its local image features and their relations a graph is used. The graph is composed of a set of nodes \( N = \{ n_1, \ldots, n_N \} \) each node is labeled with a position \( \vec{x}_i \), that corresponds to the point in the model image, which is represented by this node, and a jet \( J(\vec{x}_i) \), which was derived at the corresponding point from the model image. Moreover the graph contains a set of edges \( E = \{ e_1, \ldots, e_E \} \). Each edge is labeled with the two nodes it connects and a length. The length is usually given by the Euclidean distance between the positions of two nodes in the model image.

2.3.2 Matching

The correspondence between a model and an image is established by evaluating the elements of the mapping set \( F \), which is partly defined by the structure of the given graph and partly by general assumptions about image transformations. To facilitate the evaluation process the elements of \( F \) are assumed to be concatenations of elements of basic mapping sets.

\[
F = \{ f | f = f_m \circ \ldots \circ f_1 \} \quad \text{with} \quad f_i \in F_i \quad \forall 1 \leq i \leq m
\]  

(2.11)

This way a hierarchy of mapping sets is constructed and for a suitable choice of basic mapping sets a coarse to fine approach can be used to evaluate the mappings. To clarify matters, we describe first typical basic mapping sets, and then the different search strategies within each such set.

Basic Mapping Sets

- **Translation mappings** \( F_{\text{trans}} \): \( F_{\text{trans}} \) contains all those functions \( \tilde{f} \) which map node positions \( \vec{x}_i \) to image coordinates \( f(\vec{x}_i) \) for which

\[
\tilde{f}(\vec{x}_i) = \vec{x}_i + \vec{t} \quad \text{with} \quad t_x \in [x_{\min}, x_{\max}] \quad t_y \in [y_{\min}, y_{\max}]
\]

(2.12)

is true for all nodes.

- **Scale mappings** \( F_{\text{scale}} \): Mappings belonging to \( F_{\text{scale}} \) fulfill

\[
\tilde{f}(\vec{x}_i) = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \vec{x}_i + \vec{t} \quad \text{with} \quad s_x \in [s_{x_{\min}}, s_{x_{\max}}] \quad s_y \in [s_{y_{\min}}, s_{y_{\max}}]
\]

(2.13)

for a given \( \vec{t} \). In some applications it is also helpful to require \( s_x = s_y \).
2.3. GRAPH MATCHING

- **Local mappings** $F_{local}$: $F_{local}$ contains mapping functions which allow the nodes to be mapped fairly independently of each other. They need to fulfill

$$\|\tilde{f}(\tilde{x}_i) - \tilde{x}_i\| < d_{local}$$  \hspace{1cm} (2.14)

$$\sum_{ij} e_{ij} \|\tilde{f}(\tilde{x}_i) - \tilde{f}(\tilde{x}_j)\| - \|\tilde{x}_i - \tilde{x}_j\| < d_{global}$$  \hspace{1cm} (2.15)

where $e_{ij}$ is an indicator function which is 1 if node $i$ and $j$ are connected by an edge and 0 otherwise. The parameters $d_{local}$ and $d_{global}$ control the amount of deformation of the graph created by the mapping. $d_{local}$ limits the change for each node independently, whereas $d_{global}$ limits the total amount of deformation, depending on the graph structure.

Search Strategies

To retrieve the correct or a close to correct mapping from a basic mapping set two search strategies are employed.

- **Scanning the parameter space**: Given a parameterization of the basic mapping set, a subset of mappings is derived by homogeneous sampling of the parameter space. For each mapping in the subset the resulting feature similarity is calculated, i.e.

$$s_{mapping} = \sum_{i} s\left(\tilde{J}(\tilde{x}_i), \tilde{J}(\tilde{f}(\tilde{x}_i))\right)$$  \hspace{1cm} (2.16)

Taking advantage of the smoothness of the feature similarity functions, the mapping yielding the highest similarity is assumed to be a good approximation of the correct mapping. In the case of $F_{local}$ the parameterization is given by single node translation. The global deformation maximum is enforced by discarding a mapping if its edge distortion exceeds $d_{global}$.

- **Approximating disparities**: As mentioned in section 2.2.3 the similarity function $s_{disp}$ yields an estimated disparity $\tilde{d}_i$ for each node $i$ in addition to a similarity value. This is used to determine a mapping from a basic set by picking the mapping that minimizes

$$\sum_{i} \left(\tilde{x}_i + \tilde{d}_i - \tilde{f}(\tilde{x}_i)\right)^2 .$$  \hspace{1cm} (2.17)

Optionally the mapping found this way can be compared against the identity mapping $\tilde{f}(\tilde{x}_i) = \tilde{x}_i$. If the similarity of the neutral mapping is highest, it is chosen to be the correct mapping.
CHAPTER 2. ELASTIC GRAPH MATCHING

Moves

The combination of a basic mapping set, a search strategy, and a similarity function is called a move. A move can thus select one of a set of basic mappings. To solve the whole correspondence problem in the way described in equation 2.11 a succession of moves must be performed. The concatenated results of those moves then constitute the resulting correspondence. Such a succession of moves is called a move schedule. Two requirements must be met for this to work. Firstly the similarity potentials must be sufficiently smooth to allow for a sensible part-wise estimation of correspondence. And secondly the successive moves should result in smaller and smaller displacements of the model nodes, to allow for a coarse to fine approach with increasingly sensitive similarity functions and increasingly detailed mapping sets.

2.3.3 Recognition

Once the correspondence between a number of graphs of different objects and an image are established, recognition, i.e. the decision which object is actually present in the image, can be reduced to a simple image feature comparison. Each graph is assigned a similarity, which is calculated as the average similarity of the jets of each graph node and its corresponding image jet. The graph with the highest similarity is now assumed to represent the object most likely shown in the image.

2.4 Bunch Graph Matching

So far the correspondence problem was viewed as associating two image points with each other, that refer to an identical point in the physical world. This requires a model and an image to be derived from the exact same object. For many applications it would advantageous to relax this constraint, such that the correspondence between two different objects of the same kind could be established. It would be even more desirable to represent a whole class of objects with one single model.

For certain object classes this is achieved by bunch graph matching (Wiskott, 1995; Wiskott et al., 1997), an extension to elastic graph matching. Bunch graph matching relies on the idea that objects within one class have common fiducial points, so called landmarks, that can be identified for each member of the class, and further that these fiducial points have approximately identical geometrical relations for all members and only differ in their appearance.

2.4.1 Features and Similarities

To cover variation in point appearance bunch graph matching uses a bunch of jets to describe a single feature point. So each node is labeled with a set of jets derived from
a number of examples of the object class to be represented. All other aspects of elastic graph matching remain unchanged, apart from the similarity functions. The similarities are evaluated as nearest neighbor similarities, i.e.

\[
\begin{align*}
    s_{\text{abs}}^{\text{bunch}} &= \max_i s_{\text{abs}}(\vec{J}_{\text{bunch}}^i, \vec{J}_{\text{image}}) \\
    s_{\text{phase}}^{\text{bunch}} &= \max_i s_{\text{phase}}(\vec{J}_{\text{bunch}}^i, \vec{J}_{\text{image}}) \\
    s_{\text{disp}}^{\text{bunch}} &= \max_i s_{\text{disp}}(\vec{J}_{\text{bunch}}^i, \vec{J}_{\text{image}})
\end{align*}
\]  

(2.18)  

(2.19)  

(2.20)

where the \( \vec{J}_{\text{bunch}}^i \) are all the jets constituting a bunch. In the same spirit the disparity is calculated using the jet from the bunch, which yields the highest similarity.

\[
\begin{align*}
    \vec{d}_{\text{bunch}} &= \vec{d}_{\text{image}}^{i_{\text{max}}}  \\
    i_{\text{max}} &= \arg \max_i s_{\text{disp}}(\vec{J}_{\text{bunch}}^i, \vec{J}_{\text{image}})
\end{align*}
\]  

(2.21)  

(2.22)

This way the correspondence problem can be solved for classes of objects without relying upon a representation in terms of a model graph for each single object. It has proven very successful in tasks such as face and gesture recognition (Wiskott, 1995; Wiskott et al., 1997; Triesch and von der Malsburg, 1997; Triesch, 1999; Phillips et al., 1999).

2.4.2 Recognition

One major advantage of bunch graph matching over elastic graph matching is that recognition of a specific member of a class is decoupled from solving the correspondence problem. Once the correspondence is established by a class specific bunch graph, the similarities needed for recognition can be obtained by simple comparison of one set of local image features, which is only once extracted from the image, and all available models. This way matching time does not scale with the number of models any more, thus reducing recognition time considerably.

2.5 Discussion

After having laid out the basics of elastic graph matching, it is instructive to see how much knowledge about the world and any represented object is actually contained in the approach, and how it is obtained.

Elastic graph matching and its extension, bunch graph matching, are built upon the notion that objects are described by two basic qualities, visual appearance and transformation properties, which have to be treated separately. Visual appearance is what is encoded in the local features and the relative positions as represented by the graph. It is a property that is fairly easily picked up from an example image, although segmentation
of the object and fiducial point alignment, as needed by bunch graph matching, might require some manual aid.

The transformation properties of an object on the other hand are not as easily derived from an image. To apply graph matching for a given object or object class the set of possible transformations, i.e. the mapping set \( F \), must be given together with a hierarchy of degrees of freedom that allows to decompose \( F \) into basic mapping sets. As this information seems hard to get, elastic graph matching assumes that general object independent transformations such as translation and scale are dominant and can be dealt with one by one, and that all object specific transformations are small and can be dealt with by a local relaxation of the graph. Model knowledge is here only introduced in terms of the graph’s edges, which code for the mutual interdependence of nodes.
Chapter 3

Local Transformation Properties

This chapter deals with the transformation properties of views, as they were described in the context of mapping sets in the previous chapter. Such local transformation properties are at the heart of elastic graph and bunch graph matching as they restrict the task of retrieving the correspondence between two views of an object to the evaluation of a predefined set of mappings. The chapter is structured according to three basic questions regarding the nature of these transformation properties.

1. Can they be derived without detailed knowledge about the object at hand?

2. Can the derived transformation properties be effectively employed to solve the correspondence problem in different tasks such as recognition and tracking?

3. What do the transformation properties and their associated mapping sets reveal about the state of the physical world?

3.1 Transformation Properties

In order to determine the local transformation properties, we will resort to bunch graphs as basic model representation. Bunch graph matching has proven to find reasonable solutions to the correspondence problem, and is therefore ideally suited to serve as an initial ad hoc representation. The detected correspondences can then be used to retrieve mapping sets which capture the properties of typical object specific transformations.

In this chapter we will limit ourselves to the exemplary case of faces, because bunch graph matching has been extensively applied to this object class and face recognition as well as face understanding constitutes an especially interesting problem, with a wide range of possible applications.
3.1.1 Generating Data

In order to retrieve mapping sets which represent the typical transformations of faces a large database of face images is labeled with facial landmarks via bunch graph matching. The employed bunch graph consists of 48 nodes, each labeled with a bunch of 38 jets taken from 38 different persons. This bunch graph contains the only face specific information available to the system. The landmarks are now determined by a two-step move schedule.

1. **Scan translation move with \( s_{abs} \):** To determine the rough location of the face, the translation mapping set is homogeneously sampled on a grid with 2 pixels inter-node distance. The resulting subset is evaluated by the \( s_{abs} \) similarity function, which exhibits the smoothest similarity potential (see figure 2.2).

2. **Scan local move with \( s_{g} \):** The individual landmark positions are determined by homogeneous sampling of the local mapping set on a \( 20 \times 20 \) grid for each node, with \( d_{local} = 20 \) and \( d_{global} = \infty \) (see section 2.3.2), and subsequent evaluation of this mapping subset according to the \( s_{g} \) similarity function. The choice of \( d_{global} \) allows to determine the individual node correspondences independent of each other.

Of course a more elaborate matching schedule might lead to better results, if for example scaling, rotation in plane or in depth would be compensated for explicitly. But the matching is kept this simple, in order to keep the system unbiased against possible transformations. The matching schedule is based on only two basic assumptions – translation invariance and localized deformations. The notion that objects can appear anywhere in the field of view is not special to faces and can safely be assumed valid under most circumstances. The need to compensate for translation is thus explicitly acknowledged in the matching schedule. The second assumption is that all remaining transformations can be expressed in terms of local node translations. In the above schedule those node translations are confined to a \( 20 \times 20 \) pixels square. This assumption is of course violated in cases where a view differs strongly from the model views. This imposes restrictions on the range of transformations the system can deal with. It is none the less required in order to resolve basic ambiguities resulting from a face’s symmetry.

This way the correspondences between the bunch graph and 1000 images of approximately 800 individuals were retrieved. The images were \( 128 \times 128 \) gray value images taken from the FERET database supplied by the ARL. The gallery contained only faces of approximately the same size and only those head poses were included which were labeled as frontal by the ARL. This reduced pose variations to approximately \( \pm 15^\circ \). Some results are shown in figure 3.1. The correspondences are of course far from perfect. Due to the large and fairly unconstrained search space landmarks are often misplaced. But one could maintain the hope that the data, although noisy, is sufficient to allow extraction of some general transformation properties.
3.1. Transformation Properties

Figure 3.1: Example of automatically labeled faces: Displayed are 10 arbitrarily chosen examples of a set of approximately 1000 images. The retrieved correspondences are displayed by superimposing the mapped graph.

3.1.2 Statistical Analysis

For further analysis each correspondence mapping is assigned to a point in a Euclidean space, whose dimensionality is twice the number of nodes in the bunch graph, in this case 96. The coordinates of those points are given by the coordinates of the mapped landmark positions, i.e. the point $\vec{X} = (x_1, y_1, \ldots, x_{48}, y_{48})^T$ is associated with a correspondence mapping, which maps the $i$-th landmark to the position $(x_i, y_i)^T$ in the image for all $i \in \{1, \ldots, 48\}$.

Later it will be necessary to extract the presumed position of a single landmark $m$ in a given image from the correspondence mapping. To this end, the simple projection operator $D_m$ is defined.

$\vec{x}_m = (x_m, y_m)^T = D_m \vec{X}$ \hspace{1cm} (3.1)

This way a point in the 96-dimensional space is assigned to each image in the gallery, based on the retrieved correspondences. Then, principal component analysis (PCA) is performed on this point distribution. PCA was used for several reasons. Firstly, it was demonstrated in (Lanitis et al., 1997) for a similar set of data, that it yields good results, at least as long as the correspondences are perfect and the database is chosen carefully to reflect an appropriate range of transformations.

Secondly principal component analysis decomposes the input space into linear subspaces, which reflect the variation within the dataset to different degrees. This is, as we will see, closely related to the notion of decomposing the correspondence mappings into of a hierarchy of basic mappings sets, as used in elastic graph matching (see section 2.3.2). By identifying those linear subspaces with the basic mapping sets improvements in terms
of object specific basic mapping sets are easily incorporated into the graph matching framework.

The principal components are calculated from the covariance matrix $C$ of the data.

$$C = \frac{1}{N-1} \sum_{n=1}^{N} (\tilde{X}_n - \tilde{M}) (\tilde{X}_n - \tilde{M})^T, \text{ with } \tilde{M} = \frac{1}{N} \sum_{n=1}^{N} \tilde{X}_n$$

(3.2)

$N$ is here the number of correspondence mappings considered. Exploiting the symmetry of the covariance matrix $C$, it can be written as

$$C = P \begin{pmatrix} p_1 & 0 \\ \vdots & \ddots \\ 0 & p_{96} \end{pmatrix} P^{-1}$$

(3.3)

with

$$P^T = P^{-1}$$

(3.4)

$$p_1 \geq p_2 \geq \cdots \geq p_{96}.$$ 

(3.5)

The principal components $\tilde{P}_i$, the column vectors of $P$, are thus normalized, mutually orthogonal and sorted in terms of their corresponding eigenvalues. They form a basis of the correspondence mapping space such that the mean squared reconstruction error

$$E(K) = \frac{1}{N-1} \sum_{n=1}^{N} \left| (\tilde{X}_n - \tilde{M}) - \sum_{k=1}^{K} \left( (\tilde{X}_n - \tilde{M})^T \tilde{P}_k \right) \tilde{P}_k \right|^2 = \sum_{k=K+1}^{96} p_k$$

is minimal for any given $K \in \{1, \ldots, 96\}$ (a more detailed account of PCA can be found in section A of the appendix).

### 3.1.3 Results

Every mapping $\tilde{X}$ can now be expressed as a superposition of basic mappings, represented by the principal components $\tilde{P}_k$.

$$\tilde{X} = \tilde{M} + \sum_{k=1}^{96} a_k \tilde{P}_k$$

(3.7)

In order to visualize the first few resulting principal components which according to equation 3.6 represent the most typical transformations, exemplary model graphs were generated, see figure 3.2. These examples were generated by mapping the model graph according to $\tilde{X}_{i\text{\scriptsize{example}}} = \tilde{M} + c_i \tilde{P}_i$, where the $c_i$ were arbitrarily set to $c_i = -4\sqrt{p_i}$ with $i \in \{1, \ldots, 6\}$. To better see what kind of correspondence mappings the principal
Figure 3.2: **Principal Components of Face Graphs**: The first row shows the mean graph given by the mean positions of each node as determined by the automatic landmark finding. The second and third rows show those graphs which differ from the mean graph by 4 standard deviations along the first to sixth principal component respectively. It seems that the first two components code translation, the third and fifth component scale, and the fourth and sixth component seem to be related to rotation in depth. Components beyond the sixth could not be interpreted by visual inspection.

components imply, another from of visualization was chosen, see figure 3.3. Here the mapping of the individual landmarks was interpolated linearly to allow a mapping of all gray values from an example image according to the prototypical mappings. To this end a graph was placed manually on an example image. The graph was then mapped along the first six principal components for $c_i \in \{-4\sqrt{p_i}, -2\sqrt{p_i}, 0, 2\sqrt{p_i}, 4\sqrt{p_i}\}$ and the gray values were mapped according to a linear interpolation of the landmark mapping based on a Delaunay triangulation of the graph.

As it turns out the principal components are readily interpretable. They seem to
Figure 3.3: **Textured Principal Components of Face Graphs**: To allow for easier interpretation the graphs of typical deformations are illustrated in terms of the mapping on the gray value image they imply. A picture of a face was manually labeled with a graph. A Delaunay triangulation was performed on the graph. The graph was then deformed according to the derived principal components as in figure 3.2. The pixel values within each triangle of the graph were mapped linearly onto the deformed graph. The columns show deformations according to the principal components one to six respectively, while the rows show deviation from the mean of -4,-2,0,2 and 4 standard deviations.
3.1. TRANSFORMATION PROPERTIES

code transformations that are easily identified and named by visual inspection. The first two are translation, then there is scaling and rotation in depth. This is remarkable for several reasons. Firstly we started off with a very noisy database of automatically determined correspondences. Although the database contained a lot of different individuals and was restricted to approximately frontal pose, the inter-individual variations (such as for example jaw size over eye distance) are not dominant. The main variations seem to stem from geometrical variations. The only inter-individual variation visible in the first six components is expressed in the independence of scaling in x- and y-direction, which might be due to different head shapes. Although no explicit or implicit knowledge on the three-dimensional transformations of rigid object was provided, their main properties were captured. And the degrees of freedom are nicely separated in an intuitive fashion, namely into translation, scale, and rotation.

This supports the notion, that faces are a nice or linear object class (Beymer and Poggio, 1996; Vetter and Poggio, 1997). Which means that the transformation properties of individual faces are essentially the same and can be generated from a set of prototypes for which the transformation properties are known. This in turn is a reasonable explanation of why most successful face recognition systems are based on texture analysis rather than the analysis of geometrical feature relations. The systems tested in the FERET test for example all relied on some sort of texture analysis (Phillips et al., 1999). The inter-individual changes of geometrical relations are always superimposed with those stemming from a head’s geometrical transformations in space. As the latter seem to feature more prominently in the statistics of frontal faces, they have to be compensated for explicitly and with great care, before inter-individual changes can be evaluated. This is often very difficult. One geometrical relation that might be easily picked up and is also strongly reflected in the appearance of texture features, such as jets, is the head’s aspect ratio, which was the only inter-individual geometric feature prominently present in the found principal components.

On this basis one could speculate that psychophysical experiments showing close to equal recognition performance of human subjects for shape and texture normalized facial images (O’Toole et al., 1998) are mainly influenced by this aspect ratio. If this was indeed so, one would assume that the performance for recognizing the shape normalized faces would dramatically increase, if the shape normalization procedure would preserve the aspect ratio.

Although a very simple matching schedule was employed to solve the correspondence problem, which contained no explicit compensation of transformations apart from translation, the results of the PCA are also consistent with the hierarchy of basic mapping sets usually employed by elastic graph matching (see section 2.3.2). The ranking of the derived principal components seems to justify the ad hoc approach of decomposing the correspondence problem into a search set of hierarchical organized basic mapping sets. The retrieved principal deformation thus seem to be a property of the database. But as no special effort was made when assembling the database of close to frontal faces to
include certain transformations, as for example in (Lanitis et al., 1997), it is likely that the retrieved principal deformations are also a property of frontal faces in general.

3.2 Recognition with Object Knowledge

The Principal Components derived from an initially ignorant object recognition system provide us with a hierarchy of linear subspaces. The mappings associated with these subspaces can be classified into more or less typical ones, depending on the amount of variance they cover on our training set. If this ranking is not just a quality of the database, but a property of faces in general or at least a property of this special representation of faces, this hierarchy can be used to constrain the correspondence problem in a number of scenarios. The number of degrees of freedom can be limited in a straightforward fashion, resulting in lower computational complexity and fewer ambiguities. In the following this will be demonstrated for face recognition by incorporating basic mapping sets into the move schedule of bunch graph matching, which are a direct consequence of the statistical analysis carried out above.

3.2.1 Experiments

The influence of mapping constraints on face recognition performance was tested for three different sets of constraints and four different face galleries containing different types of variations. The four galleries each contained the faces of 110 persons. The persons were the same for all galleries. The galleries were distinct from the gallery originally used to derive the principal components in the sense that they contained different persons and were recorded under different conditions. The first gallery contains expressionless frontal faces, the second expressionless faces rotated by approximately 15° in depth, the third expressionless faces rotated by approximately 30° in depth, and the fourth frontal faces with varying expressions. Three different move schedules were used to solve the correspondence problem with the bunch graph described above.

1. no constraints: The move schedule is identical to the one described in section 3.1.1, it thus makes virtually no assumptions about the transformation properties of faces.

2. manually optimized constraints: This move schedule serves as an example of a typical move schedule as employed, e.g., in the FERET test (Phillips et al., 1999). The decomposition into basic mapping sets and the associated search strategies were optimized manually. The move schedule is given by the succession of a scan translation move, a scan scale move covering five scales in the range of 0.8 to 1.2 and a disp scale move. Lastly, a disp local move is performed to compensate for unknown object specific deformations. All moves employ the $s_{abs}$ similarity function apart from the disp moves, which rely on $s_{disp}$.
3.2. RECOGNITION WITH OBJECT KNOWLEDGE

3. PCA-based constraints: Translation invariance is again achieved by an initial scan translation move with the $s_{abs}$ similarity function. All other transformations are compensated by a disparity move on a mapping set constrained to the space spanned by the first six principal components. At each graph node the disparity is estimated. The resulting disparity field $\tilde{D}$ is then projected upon the first six principal components to yield the new constrained disparity field

$$\tilde{D}' = \sum_{n=1}^{6} (\tilde{P}_n^T \tilde{D}) \tilde{P}_n.$$  (3.8)

The thus constrained disparity $\tilde{D}'$ defines the new basic mapping. It must be noted that in contrast to the disp local move an increase in similarity was not required to validate the results, because the move was sufficiently constrained to render the additional check superfluous.

The performance was measured in percent of correct recognitions for a given pair of galleries. For each such pair one gallery is chosen to be the probe gallery, which contains the “unknown” images presented to the system. The system’s tasks is now to pick the most similar image from the other gallery, the data gallery. Whenever the presented image from the probe gallery stems from the same person as the image with highest similarity from the data gallery, recognition is said to be achieved.

3.2.2 Results

The results for all pairs of galleries and matching schemes are shown in the tables below in percent of correct recognitions.

<table>
<thead>
<tr>
<th></th>
<th>no constraints</th>
<th>manually optimized constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>probe</td>
<td></td>
</tr>
<tr>
<td></td>
<td>frontal</td>
<td>15°</td>
</tr>
<tr>
<td>data</td>
<td>frontal</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>15°</td>
<td>86</td>
</tr>
<tr>
<td></td>
<td>30°</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>expression</td>
<td>83</td>
</tr>
<tr>
<td>data</td>
<td>frontal</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>15°</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>30°</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>expression</td>
<td>82</td>
</tr>
</tbody>
</table>
Figure 3.4: **Landmark Finding**: Shown are two examples of correspondence finding with different constraints. **A** displays the *free* scenario, where the feature points are treated as fairly independent. **B** displays the *manually optimized* version. Here mainly translation and scale are compensated for explicitly. Every other deformation is taken care of by a *local move*. **C** shows matching with the constraints derived from the *PCA* approach.

<table>
<thead>
<tr>
<th align="left"><strong>PCA constraints</strong></th>
<th align="left">probe</th>
</tr>
</thead>
<tbody>
<tr>
<td align="left"></td>
<td align="left">frontal</td>
</tr>
<tr>
<td align="left">data</td>
<td align="left">100</td>
</tr>
<tr>
<td align="left">15°</td>
<td align="left">88</td>
</tr>
<tr>
<td align="left">30°</td>
<td align="left">71</td>
</tr>
<tr>
<td align="left">expression</td>
<td align="left">83</td>
</tr>
</tbody>
</table>

In the comparison of frontal face with and without expressions the nature of constraints hardly registers in the recognition performance. For small changes in expression, such as smiling compared to no expression at all, the recognition is fairly easily achieved. In such cases it was shown before that the nature of constraints is irrelevant for recognition performance (Wiskott, 1999). For more challenging changes in expression, where the appropriate constraints might yield an advantage, the matching schedules do not differ that much, as none of them compensates for these transformations explicitly.

As soon as rotation in depth is included in one of the galleries the improved correspondences (see figure 3.4) yielded by the PCA-constrained matching increase the recognition
performance significantly. The improvement is especially large if compared to the ad
hoc constraints, that usually accompany elastic graph matching. It seems that ill-fitted
constraints are worse than no constraints at all. As the manually optimized constraints
do not take rotation in depth into account, they prohibit a correct determination of the
correspondences, whereas the ambiguities in the given images were not severe enough
to show the drawbacks of an unconstrained matching. Although this explains the poor
performance of the ad hoc constraints, the question remains how to find a good set of
constraints if those used do not sufficiently reflect the object properties. We have shown
that the constraints easily bootstrapped from the unconstrained case via principal com-
ponent analysis constitute a valid answer. On the one hand they are more accurate than
the unconstrained matching and on the other hand as in the case of the manually opti-
mized constraints they are efficient in terms of computation time as they do not require
an extensive search in image space.

3.3  Tracking with Object Knowledge

Another area where constraining the correspondence problem might lead to significant
improvement is the tracking of feature points. The applicability of our approach to this
task is explored in this section. We will start with a short introduction to a feature
point-based tracking scheme. Using this scheme two methods of taking advantage of
a constrained correspondence mapping space are developed and their performances are
evaluated.

3.3.1  Tracking Single Points

Disparity estimation

Tracking a point from one image to the next requires the estimation of its movement or
disparity between those two images, which again requires a solution to the correspondence
problem. But as the disparity is usually very small, no fullfledged matching is needed.
The disparity is rather evaluated in terms of phase differences of single jets. The phases
of the complex-valued results of a Gabor transformation of an image were used by Maurer
(Maurer and von der Malsburg, 1996b) and McKenna et al. (McKenna et al., 1997) to
track image features from one frame to another in a sequence of images. The method is
based on the findings of Fleet and Jepson (Fleet and Jepson, 1990) and Theimer and Mal-
lot (Theimer and Mallot, 1994), who used this method to recover the binocular disparity
from pairs of stereo images. The underlying idea is that the phase difference between the
two complex valued results of a Gabor filtering at two points in an image varies approxi-
mately linearly with the distance of those points. Extracting two jets at positions $\vec{x}$ and
$\vec{x}', their mutual disparity can be calculated by maximizing the similarity function $s_\delta$ (see
section 2.8) or rather, to simplify things, the first terms of its Taylor expansion in the
disparity $\tilde{d}$.

$$s_{\text{disp}} = \sum_k a_k(\tilde{x})a_k(\tilde{x}')(1 - 0.5(\phi_k(\tilde{x}) - \phi_k(\tilde{x}') - \tilde{k}^T \tilde{d})^2)$$

(3.9)

This similarity function has one unique maximum, which can easily be determined by standard techniques, but this approximation only holds for small disparities.

To exploit the multiscale properties of the Gabor transformation, the different frequencies and kernel sizes in the spatial domain have to be accounted for. To this end the disparity is first estimated using only a subset of lower frequency levels. The number of levels used in this initial step is called focus. After this initial step more and more frequency levels are taken into consideration. In each recursion one additional level is considered and the corresponding phase differences are corrected by $2\pi$, so that the phase differences $\Delta\phi_k = \phi_k(\tilde{x}) - \phi_k(\tilde{x}')$ are always confined to the interval

$$\tilde{d}^T \tilde{k} - \pi < \Delta\phi_k \leq \tilde{d}^T \tilde{k} + \pi.$$  (3.10)

Here, $\tilde{d}$ is the current estimate of the disparity as evaluated by the lower frequency levels. This way the estimated disparity at a certain level can maximally differ by half a wavelength from the estimate of the lower frequency levels. Thus the lower frequencies resolve the $2\pi$ ambiguity for the higher frequencies.

In case the estimated intermediate disparity exceeds the so called relevance radius of the next higher frequency level, the process is terminated. A level’s relevance radius is usually expressed as a multiple of $\sigma_{\text{eff}}$:

$$\text{relevance radius} = c\sigma_{\text{eff}} = c \frac{\sigma}{|k|}.$$  (3.11)

The difference in phase between two jet coefficients is thus ignored if it can be assumed that their associated filters are not sufficiently overlapping in image space. This way the locality of the Gabor transform is taken into account. If a reliable estimate of the similarity is required, as for example in the elastic graph matching scenario discussed in section 2.2.3, the relevance radius cannot be applied, because the average similarity between randomly chosen jets is a function of the number of jet components considered. Ignoring certain jet components thus leads to incomparable similarities as shown by (Maurer, 1999).

In the following considerations regarding point tracking the parameters are set to

$$c = 2, \text{ focus} = 1$$  (3.12)

The parameters of the Gabor transform are chosen as described in section 2.2.2.

**Tracking**

This disparity estimation is a very general method, which was used to calculate stereo disparity maps (Theimer and Mallot, 1994), extract flow-fields (Wieghardt, 1996; Wiskott,
3.3. TRACKING WITH OBJECT KNOWLEDGE

1997) and solve the correspondence problem in object recognition (Würtz, 1995; Wiskott, 1995). In order to turn the disparity estimation into a method for point tracking the approach of (Maurer and von der Malsburg, 1996a; Maurer, 1999) is adopted.

In the usual point-tracking scenario a set of images or frames $I_0, \ldots, I_n$ are given, where all images are recorded by the same camera at different points in time $t_0, \ldots, t_n$. These points in time usually are approximately equidistant ($\Delta t \approx t_{i+1} - t_i$ for all $i$) and $\Delta t$ is assumed to be sufficiently small for the sequence to encode a smooth motion. Moreover, a set of starting coordinates $\bar{x}_1(t_0), \ldots, \bar{x}_m(t_0)$ is given, which correspond to points in image $I_0$ and define the points to be tracked. One is now interested in the corresponding point coordinates $\bar{x}_1(t_i), \ldots, \bar{x}_m(t_i)$ in image $I_i$ at any given moment in time $t_i$. These coordinates are obtained by a three-step process.

1. extract jets $J_i(\bar{x}_1(t_i)), \ldots, J_i(\bar{x}_m(t_i))$ at the current positions in image $I_i$.

2. calculate disparity to the jets extracted from the next image $I_{i+1}$ at the same image-coordinates $\bar{d}_n = \bar{d}_n(J_i(\bar{x}_n(t_i)), J_{i+1}(\bar{x}_n(t_i)))$.

3. calculate new positions in image $I_{i+1}$: $\bar{x}_n(t_{i+1}) = \bar{x}_n(t_i) + \bar{d}_n$

4. proceed with step one.

A more detailed description is given in (Maurer, 1999), where the problem of extracting jets at subpixel positions is also addressed.

The method described above has a number of advantages. Firstly, it does not require any prior knowledge of the object or point to be tracked. It is thus very easily applied to large range of problems, from tracking facial features over tracking heads to tracking any part of any object. Secondly, the method is based on one single feature comparison, no search in image space for the correct correspondence is required. Thirdly, the method comes with a natural measure of confidence of its own estimates, given by the similarity between the two jets (see equation 3.9), which can be used to determine whether the tracking has failed and new initialization is necessary (see for example (Potzsch, 1999)). And lastly, it is based on features that are well known and understood, and valued for their performance in computer vision problems as well as their psychological and biological relevance.

Background Suppression

This amount of simplicity and flexibility comes at a price. Problems occur mainly for points which are close to an object’s boundary. Here background motion, if different from the object motion, may distort the estimated disparity significantly. This is because a jet at a boundary codes nearly as much background as object information. For some applications, such as in facial feature tracking, this presents no problem, because the points of interest are safely embedded within the object. But for some applications it is
a major source of problems, because the points of interest or even the only well defined points of an object are on its boundary. Existing methods devised to compensate for this effect, e.g. (Pötzsch, 1994), require detailed knowledge of the object boundary and thus reduce the flexibility of the approach. One easy way of reducing the impact of this problem is breaking the symmetry in treatment of the model and the reference jet, the former being the one taken from a previous image and the latter the one from the current, and let the model jet accumulate some information of what is actually being tracked and what not. A simple way of suppressing the background is to blur it and thus reduce its influence on the jet coefficients. As we are only interested in reducing the impact of those parts of the image that do not move the same way as our point of interest, we propose a modification to the original approach of (Maurer and von der Malsburg, 1996a; Maurer, 1999), where simple motion-blur is used, thus avoiding explicit segmentation.

To this end the extraction of the model-jet must be changed. The model jet is no longer extracted from the current image but from a linear combination of previous images, while aligning the image coordinates such that the mutual disparity of our point of interest in all images is zero. So the image, from which the model jet is extracted at position $\mathbf{x}_0(t_i)$ is given by the recursive formulation

$$I_{i}^{\text{sum}}(\mathbf{x}) = (1 - \alpha) I_{i-1}^{\text{sum}}(\mathbf{x} - \mathbf{x}_0(t_i) + \mathbf{x}_0(t_{i-1})) + \alpha I_i(\mathbf{x}).$$  \hspace{1cm} (3.13)

The constant $\alpha \in [0, 1]$ determines how flexible the model jet is. The larger $\alpha$ the faster the model jet adapts to changes in the appearance for the point of interest. Figure 3.5 illustrates how the background is blurred and the point of interest is slowly defined. This simple extension increases robustness, because small tracking errors can be compensated for in subsequent frames, and decreases sensitivity to background motion. In all experiments carried out these advantages outweighed the disadvantage of decreased adaptivity. To employ the above for multiple points of interest, without increasing the computational expense unnecessarily, the linearity of the Gabor transform is used to yield the following tracking process.

1. extract jets $\tilde{J}_i(\mathbf{x}_1(t_i)), \ldots, \tilde{J}_i(\mathbf{x}_m(t_i))$ at the current positions in image $I_i$.
2. update model jets $\tilde{J}_i^{\text{model}}(\mathbf{x}_n(t_i)) = (1 - \alpha) \tilde{J}_{i-1}^{\text{model}}(\mathbf{x}_n(t_{i-1})) + \alpha \tilde{J}_i(\mathbf{x}_n(t_i))$
3. calculate disparity to the jets extracted from the next image $I_{i+1}$ at the same image-coordinates $\tilde{d}_n = \tilde{d}_n \left( \tilde{J}_i^{\text{model}}(\mathbf{x}_n(t_i)), \tilde{J}_{i+1}(\mathbf{x}_n(t_i)) \right)$.
4. calculate new positions in image $I_{i+1}$: $\mathbf{x}_n(t_{i+1}) = \mathbf{x}_n(t_i) + \tilde{d}_n$
5. proceed with step one.

Figure 3.6 shows the differences to the former approach for an artificial example.
3.3. TRACKING WITH OBJECT KNOWLEDGE

3.3.1 Tracking Constrained Groups of Points

Following our main line of thought, we now return to the application of object knowledge in terms of mapping sets. The question is how to integrate knowledge about an object’s transformation properties into the above described tracking scheme. Two approaches are discussed below.

External Constraints

Given a set of constraints on the possible disparities $\vec{d}_i$ of all points $i$ of the form

$$\vec{d}_i - \vec{f}_i(\vec{\varepsilon}) = 0,$$  \hspace{1cm} (3.14)

where $\vec{\varepsilon}$ are the underlying model parameters, a three-step process can be used. First the disparities are estimated as before, assuming all nodes to be mutually independent. Next the constrained disparity configuration is calculated that is closest, in a least square sense, to the estimated disparities. The disparities are subsequently changed to those given by the constraints.
(a) **tracking examples**: A: with background suppression and B: without background suppression

(b) **maximal tracking error**

Figure 3.6: **Tracking with and without Background Suppression**: On a simple example the influence of background structure on the tracking performance was tested for the original (Maurer, 1999) and the modified method. A translating cross was to be tracked in the face of a distractor consisting of a grey value step in the background. The experiment was done for varying distractor intensities. Figure 3.6(b) shows the maximal tracking error within a sequence as function of the contrast ratio between the target and the distractor. The contrast ratio was defined as \( \frac{g_{\text{right}} - g_{\text{left}}}{g_{\text{cross}} - g_{\text{left}}} \). Where \( g_{\text{cross}}, g_{\text{left}}, \) and \( g_{\text{right}} \) were the gray values of the cross, the left and the right part of the background respectively.
3.3. TRACKING WITH OBJECT KNOWLEDGE

the constrained configuration. So a $\tilde{\varepsilon}_{\text{min}}$ has to be calculated for which

$$\sum_i \| \tilde{d}_i - \tilde{f}_i(\tilde{\varepsilon}_{\text{min}}) \|^2$$  \hspace{1cm} (3.15)

is minimal. Assuming a good estimate $\tilde{\varepsilon}_0$ of $\tilde{\varepsilon}_{\text{min}}$ is given by some source outside the actual tracking such that any deviation $\Delta \varepsilon = \tilde{\varepsilon}_{\text{min}} - \tilde{\varepsilon}_0$ is small, the resulting equation can be linearized by taking only the first order Taylor expansion into account. Assuming that the $\tilde{f}_i$ are differentiable and in general well behaved, an approximate solution of (3.15) can be obtained by solving:

$$\sum_i \left( \frac{\partial \tilde{f}_i}{\partial \varepsilon} \right)_{\tilde{\varepsilon}_0}^T \left( \frac{\partial \tilde{f}_i}{\partial \varepsilon} \right)_{\tilde{\varepsilon}_0} \Delta \varepsilon = \sum_i \left( \frac{\partial \tilde{f}_i}{\partial \varepsilon} \right)_{\tilde{\varepsilon}_0}^T (\tilde{d}_i - \tilde{f}_i(\tilde{\varepsilon}_0))$$  \hspace{1cm} (3.16)

Disparities which fulfill the constraints are then given by

$$\tilde{d}_i = \tilde{f}_i(\tilde{\varepsilon}_0 + \Delta \varepsilon).$$  \hspace{1cm} (3.17)

If the points are simply constrained to an $N$-dimensional linear subspace, as derived by the principal component analysis in section 3.1.2, the expressions simplify a lot due to the linearity. In this case the $\tilde{f}_i$ are given by

$$\tilde{f}_i(\varepsilon) = \mathcal{D}_i \left( \sum_{n=1}^{K} \tilde{P}_n \varepsilon_n \right).$$  \hspace{1cm} (3.18)

$\mathcal{D}_i$ is the projection operator introduced in section 3.1.2. Substituting the solution of (3.16) for these constraints into equation (3.17) yields

$$\tilde{d}_i = \mathcal{D}_i \sum_{n=1}^{K} \left( \sum_i \tilde{d}_i^T (\mathcal{D}_i \tilde{P}_n) \right) \tilde{P}_n,$$  \hspace{1cm} (3.19)

if $\tilde{\varepsilon}_0 = 0$ is assumed. This can alternatively be written as

$$\tilde{D}' = \sum_{n=1}^{K} \left( \tilde{P}_n^T \tilde{D} \right) \tilde{P}_n.$$  \hspace{1cm} (3.20)

$$\tilde{d}_i = \mathcal{D}_i \tilde{D}'$$  \hspace{1cm} (3.21)

This is exactly what was done to constrain the matching in section 3.2, where is was used for the sake of simplicity.

The main advantage of this method is the distinction between the disparity estimation and the implementation of the constraints. This independence allows any kind of approach to be applied to calculate the disparities. Moreover, the distance between constrained and unconstrained disparities provides an instant measure of the model’s accuracy. Examples
CHAPTER 3. LOCAL TRANSFORMATION PROPERTIES

Figure 3.7: The aperture problem: Whenever local features are used for disparity estimation, situations as shown in A are bound to happen. Here the correspondence of a point on a moving bar cannot be unambiguously resolved from one frame to the next, because within the region considered no information about the vertical disparity is supplied. In cases like this it is in general not a good idea to force decisions and then try to cope with errors later by regularization, because vital information on the nature of the ambiguities is lost. It would be far better to consult more features (like in B) that provide a broader perspective, if feasible.

of this kind of implementation can be found in (Broida et al., 1990; Li et al., 1993; McKenna et al., 1997).

But applying the constraints externally, i.e., after the disparities are already calculated, has some serious drawbacks concerning the aperture problem, see figure 3.7. This is even worse, because one of the reasons to consider constraints is fighting the aperture problem. The separation of model knowledge and motion estimation forces decisions while estimating the initial disparities, even if information is inadequate. This leads to symmetry breaking, which in turn causes small errors to accumulate over time and thus the total tracking result to deteriorate in quality (see figure 3.8).

Internal Constraints

For those reasons an alternative approach might be preferred. We will call this approach internal, because here the model knowledge is directly incorporated into the motion estimation. The main point is not to minimize the distance between independently moving points and constrained groups of points in terms of model parameters, but to directly maximize the feature similarity in the model parameters. Given constraints of the same sort as above (see equation 3.14) and the phase-based disparity estimation of equation 3.9, the constrained disparities can be found by maximizing

$$s(\varepsilon) = \sum_i \frac{\sum_k a_k(\tilde{x}_i) a_k(\tilde{x}'_i) \left( 1 - 0.5 \left( \phi_k(\tilde{x}_i) - \phi_k(\tilde{x}'_i) - \tilde{k}^T \tilde{f}_i(\varepsilon) \right)^2 \right)}{|\tilde{J}(\tilde{x}_i)||\tilde{J}(\tilde{x}'_i)|}. \quad (3.22)$$
3.3. TRACKING WITH OBJECT KNOWLEDGE

Assuming again that the first order Taylor expansion of $\tilde{f}_i$ is good enough for small deviations $\Delta \epsilon$ from some initial estimate $\bar{\epsilon}_0$, a good estimate can be found by substituting

$$\tilde{f}_i(\bar{\epsilon}) = \tilde{f}_i(\bar{\epsilon}_0) + \frac{\partial \tilde{f}_i}{\partial \epsilon} \bigg|_{\bar{\epsilon}_0} \Delta \epsilon$$

into equation 3.22 and maximization in terms of $\Delta \epsilon$. This yields as the equation system for $\Delta \epsilon$

$$\sum_n \alpha_{mn} \Delta \epsilon_n = \beta_m$$

with

$$\alpha_{mn} = \sum_i \frac{1}{|J(\bar{x}_i)||J(\bar{x}_j)|} \sum_k a_k(\bar{x}_i)a_k(\bar{x}_j) \left( \tilde{f}_i^T \left. \frac{\partial \tilde{f}_i}{\partial \epsilon_m} \right|_{\bar{\epsilon}_0} \right) \left( \tilde{f}_i^T \left. \frac{\partial \tilde{f}_i}{\partial \epsilon_n} \right|_{\bar{\epsilon}_0} \right)$$

$$\beta_m = \sum_i \frac{1}{|J(\bar{x}_i)||J(\bar{x}_j)|} \sum_k a_k(\bar{x}_i)a_k(\bar{x}_j) \left( \tilde{f}_i^T \left. \frac{\partial \tilde{f}_i}{\partial \epsilon_m} \right|_{\bar{\epsilon}_0} \right) \left( \phi_k(\bar{x}_i) - \phi_k(\bar{x}_i) - \tilde{k}_i^T \tilde{f}_i(\bar{\epsilon}_0) \right).$$

This linear equation system is, of course, only the basic formulation of the disparity estimation. As in section 3.3.1 the focus and relevance radius must be taken care of, to exploit the multiscale properties of the Gabor transformation. This is handled the same way as in the unconstrained case, by repeatedly calculating the disparity with a growing number of levels, while adapting the phases according to the current estimate.

3.3.3 Results

Comparison on Artificial Data

First a simple experiment similar to the one carried out in section 3.3.1 was performed with artificial data, to highlight the difference between internal and external constraints in the presence of ambiguities. A grey rectangle was to be tracked across a distractor boundary, which is part of the background. The feature points are chosen to be on each corner of the rectangle as well as at the center of each edge (see figure 3.8(a)). The ambiguities arise from the aperture problem (see figure 3.7), because those features centered on the edges can, at least for high frequencies, only judge the disparity perpendicular to the edge reliably, the so called normal flow. Thus, constraints are needed to avoid distractions parallel to those edges. Figure 3.8 shows the setup as well as the result. Plotted is the maximal tracking error within a sequence averaged over all feature points as a function of the relative distractor contrast, measured exactly as in section 3.3.1. Two effects are visible. Firstly, internal constraints allow reliable tracking in a much larger range of distractor intensity. And secondly the change from distraction to no distraction is very sudden for the internal constraints. Whereas the external constraints show some distraction even in regimes where the low distractor intensity should allow completely unperturbed tracking.
Tracking with two Different Methods of Incorporating Constraints: Internal and external constraints were tested on a simple example, where the only degree of freedom was translation. The maximal tracking error was recorded for varying intensities of the right background part. In figure 3.8(b) the error is plotted over the distractor target contrast ratio. The ratio was defined, as in figure 3.6, as \( \frac{g_{\text{right}} - g_{\text{left}}}{g_{\text{square}} - g_{\text{ref}}}. \)

Figure 3.8: Tracking Examples: A: internal constraints B: external constraints
3.3. TRACKING WITH OBJECT KNOWLEDGE

Figure 3.9: Tracking of Faces: Face graphs were tracked through a set of sequences of varying length. The tracking error was recorded as the mean distance from manually labeled ground truth data. Figure 3.9(b) shows the tracking error as a function of sequence length and tracking method. The relative high baseline error can partly be attributed to errors or rather inconsistencies in the manual labeling.

Comparison on Real World Data

In order to see whether these effects are also visible in more natural and less ambiguous situations, experiments were carried out on sequences of moving heads. For several sequences of varying length graphs were manually placed on the first as well as on the last frame. Tracking was then performed and the mean deviation from the ground truth on the last frame was recorded. Figure 3.9(b) shows the results. Unconstrained tracking has the worst performance in this scenario, because single feature points, especially on the outline of the face, are too ambiguous to be tracked reliably. Also errors accumulate faster without constraints. Although internal constraints outperform external constraints over time, the difference is not as striking as in the example on artificial data. This is probably due to imperfect model assumptions, which effect both methods alike and somewhat counter the flaws of the external constraints. The model assumptions are imperfect for two reasons. By applying principal component analysis to a very heterogeneous dataset, a generic head model was created that is not adjusted to a specific head shape. Thus, the more a certain head deviates from the training set the worse the model fits. The training set also only included poses close to frontal, whereas the sequences contain views of heads...
rotated in depth of up to 45°. This, in combination with the strict linearity of the model, must lead to deviations for large rotations in depth.

### 3.4 Interpreting Data

So far the extracted principal components were only used to restrict the correspondence problem in two different scenarios. This was done by allowing deformations only in the linear subspace spanned by the first six principal components. No attention was paid to the structure within this linear subspace. The goal of this section is to relate properties of the physical object to the observed principal components. Partly this was already attempted in section 3.1.3, where the principal mappings were displayed in an intuitive fashion, but the interpretation was purely based upon visual inspection. We will try now to validate our claims with two experiments.

#### 3.4.1 Matching Principal Components to Simple Geometrical Deformations

In section 3.1.3 the derived principal components were roughly interpreted by visual inspection. Four of the first six components were associated with simple in-plane transformations, namely translation and scaling along the x and y axes. To test these assumptions the exact mapping representations for those transformations, in terms of the notation adopted in section 3.1.2, are calculated. A translation mapping $T$ is given by $T = \tilde{T} + p_{tx} \tilde{T}_x + p_{ty} \tilde{T}_y$ with the basic mappings

\begin{align*}
\tilde{T}_x &= \frac{1}{\sqrt{48}} (1, 0, 1, 0, \ldots, 1, 0, 1, 0) \\
\tilde{T}_y &= \frac{1}{\sqrt{48}} (0, 1, 0, 1, \ldots, 0, 1, 0, 1)
\end{align*}

and scaling as $S = \tilde{S} + p_{sx} \tilde{S}_x + p_{sy} \tilde{S}_y$ with

\begin{align*}
\tilde{S}_x &= \frac{1}{\sqrt{\sum_i \tilde{c}_i^2}} (c_1, 0, c_2, 0, \ldots, c_{47}, 0, c_{48}, 0) \\
\tilde{S}_y &= \frac{1}{\sqrt{\sum_i \tilde{c}_i^2}} (0, c_1, 0, c_2, \ldots, 0, c_{47}, 0, c_{48})
\end{align*}

where $\tilde{c}_i = (c_i^x, c_i^y)$ is the difference of the coordinates of node $i$ to the center of gravity in the mean graph. The basic mapping representations (equation 3.25-3.28) of translation and scaling are normalized and mutually orthogonal. They can thus be viewed as the idealized prototypical principal components for those transformations.
Figure 3.10: **Selectivity of Principal Components**: Shown are the projected lengths of assumed basic geometric deformations (scale and translation) onto the actual principal components. Displayed are only the first 24 principal components, the projected lengths onto the remaining components were negligible. Figures (a), (b), and (c) confirm the ad hoc interpretation of section 3.1.3, whereas figure (d) shows that scaling in y-direction is not coded for in terms of a single principal component, although the fifth component exhibits the strongest relation.

Whether the principal components derived in section 3.1.2 are related to these theoretical mapping representations is tested by evaluating the projection of the latter onto the principal components. Figure 3.10 shows the projected length onto the first 24 principal components for the four basic mappings. The projections onto the remaining components...
were negligible.

The assumption that the first, second, and third component code for translation in x-direction, y-direction and scaling along the x-axis respectively seems close to reality. The fifth component codes the major part of scaling along the y-direction, but some other components also contribute substantially to this deformation. Thus scaling along the y-direction is not represented as a pure state but as a mixture of different components.

### 3.4.2 Principal Components and Rotation in Depth

Although nice, the results for the in-plane transformations are not too surprising. Translation was explicitly included in the move schedule during the learning phase and scaling is a pattern picked up fairly easily from our data set at least for the nodes on the boundary, as background influences were minimal. More interesting are those principal components that seem to be related to rotation in depth. They are interesting in a number of ways. Having used an approximately frontal gallery as the training set, it is astonishing that rotation in depth still seems to be a major source of variation. Moreover, being able to derive a deformation model, without actually measuring or guessing the depth profile, is certainly attractive.

To support the claim that some information on 3D structure is encoded in the principal components in a readily accessible fashion, the following experiment was performed. Sequences of heads rotating from right to left from 0° to about 30° were recorded. Each of these sequences was 50 frames in length. Additionally two sequences of a journal cover displaying a face were taken, also 50 frames in length and showing a change in angle towards the camera from 0° to 30° and 45° respectively.

On each starting frame of those sequences a graph was placed manually, and this graph was tracked through the sequence as described in section 3.3.2. For each frame the correspondence mapping from the starting frame to the current is transformed into the coordinate system given by the principal components. The third, fourth and sixth component are plotted as function of frame number. The result is displayed in figure 3.11. The difference is obvious. Whereas the rotation of the magazine cover solely results in an increase of the component responsible for scaling in x-direction, which would be expected for a planar surface rotating along the y-axis under an orthogonal projection, the heads rotating in depth show a distinctly different behavior. Here the rotation in depth results in a steady change in the fourth and sixth component that is monotonous and appears almost linear (a clear judgment on linearity cannot be made, because no exact information on rotation angle per frame was attainable). These results strongly suggest that by decomposing the correspondence mapping into its principal components, properties connected to the object’s 3D structure can be easily inferred. Especially it seems possible to introduce a topology on the different views of a face based upon those principal components, which correctly reflects the neighborhood of the views in terms of their rotation in depth. This means that the views of the heads could be ordered
3.4. INTERPRETING DATA

Figure 3.11: **Principal Components under Rotation in Depth**: Graphs were placed on the initial frame of 4 sequences and subsequently tracked through the entire sequences. For each frame the correspondence mapping between the first and current frame is projected onto the third, fourth and sixth principal component. The evolution in time of these projections shows striking differences between the picture of a head and actual heads rotating in depth. The fourth and sixth component are clearly related to the heads’ motion in 3-D.
Figure 3.12: **Depth Related Principal Components**: A graph is tracked through a sequence, showing a man while driving a car. The fourth and sixth principal component of the correspondence mapping between the first and current frame are recorded and plotted as function over time. Some example images are centered over their corresponding frame number. The principal components seem to allow a qualitative interpretation of the image data in terms of the 3-D pose of the displayed head.

According to their pose without reference to the temporal context by simple inspection of the principal components. This is astonishing, because all this was derived from pure image statistics without any reference to the physical world.

Figure 3.12 displays the same experiment on a more natural dataset. The sequence shows the driver of a car. The fourth and sixth component are plotted over time or rather frame number. Distinct points of this curve are labeled with the actual images corresponding to those points in time. It can be seen that the decomposition into principal components allows at least a qualitative analysis of the driver’s pose.

### 3.5 Conclusion

We demonstrated that it is feasible to bootstrap knowledge about the transformation properties of a three-dimensional object, given a system such as *elastic graph matching*, which allows to retrieve approximate correspondences. A hierarchy of prototypical deformations could be derived from the data labeled by a version of elastic graph matching completely ignorant of these transformation properties. Using a heterogeneous data set
of close-to-frontal faces, it was surprising that the hierarchy is dominated by geometrical deformations and not, as one might have expected, by inter-person variations. Rotation in depth also featured prominently in this hierarchy, although the data set was created trying to suppress this degree of freedom. Moreover it was shown that the space spanned by these prototypical deformations can be used successfully to derive basic mapping sets and thus constrain the correspondence problem. Moves based upon these basic mapping sets were shown to increase the performance of correspondence finding and recognition, especially if rotation in depth was significant. It was demonstrated that a linear mapping from a parameter space to a mapping space as given by the principal components is ideally suited for integration into an existing tracking scheme. Although the mapping from parameters to deformations was strictly linear decent results were obtained even for rotations in depth which exceeded the angles present in the training dataset.

Not only could the mapping from parameter space to the correspondence mapping space be put to effective use, but the inverse, which assigns parameters to given deformations, seems to yield interesting results, which allow to relate deformations of the projected image of an object to parameters describing transformations in the physical world. The results described above seem to suggest that a topology of views based upon the associated principal components, accurately reflects the topology of views given by the underlying 3-D transformations.
Chapter 4

Applications

In this chapter two applications are presented which are exemplary in the sense that they show how the principles developed in chapter 3 can be applied in the framework of more complex tasks.

The first example can also be viewed as a motivation for the approach of calculating deformation subspaces from example images, while the second example motivates the reinterpretation of these subspaces as linear charts of a more complex non-linear deformation manifold.

4.1 Detailed Landmark Finding

The problem here is to locate facial landmarks in high resolution images (480 × 640 pixels) of close to frontal faces reliably and with high accuracy. To achieve the required accuracy two obstacles must be overcome. Firstly, the variation in the position relative to the camera, resulting in large variations in terms of translation and scaling of the faces within the images, must be dealt with. And, secondly, generalization over the variations between faces of different persons must be achieved. Both problems are already present in the face recognition scenario of section 3.2.1, but are exacerbated here due to the higher resolution. To tackle those problems a natural extension to the hierarchical approach of bunch graph matching is developed and the parameterization of the mapping functions by principal component analysis as introduced in chapter 3 is applied.

4.1.1 Hierarchical Processing

In section 2.3.2 it was explained how graph matching solves the correspondence problem by concatenating different moves to refine the mapping step by step. This can be taken further by not only varying the mapping function set and the search strategy associated with a move but also the bunch graph and the image resolution. This way the problem
can be broken down into subproblems such that the final solution is again given by the
concatenation of successive mapping functions.

For the task at hand the process is divided into three distinct steps, face finding, rough feature localization for the whole face, and detailed localization of individual feature
groups.

**Face Finding**

The first step is to locate the face in the given image and to get an initial estimate of its
size in the image. The main variations to be detected here are thus translation and scale.
As no detailed solution to the correspondence problem is required at this stage, it can be
carried out at a fairly low resolution. The image is thus downsamped to fit a 128 × 128
pixel square. On the resulting downsamped image matching is performed with 5 bunch
graphs of 16 nodes and 14 jets per node. The bunch graphs differ by a factor of \(\sqrt{2}\) in
size and the nodes are mainly located on the outline of the face, to achieve robustness
against inter-person variations.

Each of the five bunch graphs is matched onto the image with a move schedule con-
sisting of two moves, first a scan global move and secondly a scan scale move covering
three scales in the range of ±10% of the original graph size (see section 2.3.2).

The match resulting the highest similarity is assumed to be closest to the actual
mapping.

**Landmark Finding**

The image region given by the enclosing rectangle of the mapped face finding bunch
graph is cut out with a 10% margin and rescaled to 128 × 128 pixel. On this cut-out basic
landmark finding is performed with manually optimized constraints as described in section
3.2.1 with the exception that no local relaxation step is used. This way the approximate
positions of the eyes, mouth, nose and the lower part of the outline are determined.

**Facial Feature Finding**

For each facial feature to be found, i.e. the eyes, the nose, the mouth and the lower part of
the outline, a bunch graph was created with 45 example jets per node. The node numbers
varied for each facial feature (see figure 4.1). Each facial feature was also associated with
a node from the landmark finding graph. Around these associated nodes image regions of
size 128 × 128 pixel were cut out, on which matching was performed with the specialized
bunch graphs. Due to the different transformation properties the matching schedule was
different for each facial feature.
4.1. DETAILED LANDMARK FINDING

4.1.2 Parameterization of Mapping Functions

In order to find a good matching schedule it is, as mentioned before, important to know the valid mapping function set from which the actual mapping can be chosen. Therefore the typical deformations of a representing graph must be known.

The task at hand is an exemplary case where some typical deformations are obvious and can be readily incorporated into a matching schedule such as scale and translation,
CHAPTER 4. APPLICATIONS

but others are far from obvious. For example, how do the shapes of mouth and chin differ
between persons, and how can these differences be parameterized as to incorporate them
into a move?

Here the advantages of the statistical nature of the approach described in section 3.1.2
become clearly visible. From the principal component analysis of the node positions a
possible parameterization is readily available and can be employed without further knowl-
edge of the problem under consideration. As the data for the task at hand was limited
to 145 images, 45 randomly chosen training images and 100 test images, the initial naive
labeling, from which the principal components were extracted in section 3.1.1 is skipped
and PCA was directly performed on a set of manually positioned graphs. These ground
truth graphs were also used to create the bunch graphs. Because the correspondence
problem was solved manually only fairly few graphs (45) sufficed, as opposed to the 1000
graphs needed to retrieve the typical deformations from noisy automatically labeled data.

This way matching schedules were found for all facial features. The schedule for eyes,
nose, and mouth graph consisted of a scan global move, a scan scale move covering 5
scales in the range of ±20% of the graph size, a disp-scale move, and a disp-PCA move.
The dimensionality of the principal component subspace searched in the disp-PCA move
was different for each facial feature. The dimensionalities were 3 for the nose, 6 for the
mouth, and 5 for the eyes. Because the outline varied more strongly between persons the
scan scale move was modified as to scale in x and y direction separately and to cover 8
scales in the range of ±10% of the graph size. The disp-PCA move was constrained to a
3 dimensional subspace.

4.1.3 Results

This way the 100 test images were labeled. The performance of the system was measured
by measuring the mean Euclidean distance between the automatically placed landmarks
and landmarks manually positioned on the test images.

Because points on the outline are not well defined in their position parallel to the
outline, an additional performance measure was applied to them. Here only the error
perpendicular to the outline was measured. This was done by finding the two nearest
ground truth points $\vec{n}_1$ and $\vec{n}_2$ to each outline point $\vec{x}$ which was automatically placed
and measuring the distance between $\vec{x}$ and the line defined by $\vec{n}_1$ and $\vec{n}_2$, i.e.

$$error_\perp = \left\| \vec{x} - \frac{\vec{x}^T (\vec{n}_1 - \vec{n}_2)}{(\vec{n}_1 - \vec{n}_2)^T (\vec{n}_1 - \vec{n}_2)} (\vec{n}_1 - \vec{n}_2) \right\|. \tag{4.1}$$

The results are summarized in the table below.
If compared to the width of the faces of about 200 pixels and the width of the eyes of about 50 pixels on average, the achieved accuracy is noteworthy.

The table also shows the improvement made due to the parameterized deformation mapping sets. For this comparison the disp-PCA move was substituted by a disp-local move. It can be seen that the amount of improvement increases monotonously with the complexity of the graph, that is with the number of its degrees of freedom. This in turn suggests that instead of decomposing the face representation into representation of facial features, a complete face graph and its associated transformation properties should have been employed. This kind of holistic approach, however, would have required far more data than was available to us in order to determine the transformation properties.
4.2 Finger Tracking for Imitation Learning

In this section it is demonstrated how the previously discussed tracking techniques can be extended in a straightforward fashion beyond the so far demonstrated example of face tracking, in order to reliably estimate from a sequence of stereo images the trajectory of a human hand grasping an object.

This special problem arose in the context of a larger project aiming to provide a first step towards imitation learning. The goal of imitation learning is to facilitate teaching and general interaction with a robot by enabling the robot to reproduce actions performed by a teacher. One of the major bottlenecks in this ambitious endeavor is the perception part, in which the robot observes the teacher and thus acquires the information necessary for reproducing a given action. Here we concentrated on overcoming this bottleneck in the example of grip imitation, where a human teacher grasps an object from a table in front of the robot and by observing this grasping movement with a stereo camera system the robot is enabled to grasp the object all by itself in a stable fashion.

As the whole task was solved by an assortment of different building-blocks a short overview over the entire system is given before the problem of estimating the trajectory of the performed grasping movement and the special problems involved are discussed in detail.

4.2.1 Overview

The approach is basically composed of three distinct steps. Firstly, the teacher’s hand and subsequently his fingertips are located. Secondly the grasping movement is observed by tracking the fingertips through a sequence of images. And lastly, the robot imitates the grip by following the observed trajectory.

Hand and Finger Detection

Initially a simple hand tracking scheme serves as an attentional mechanism allowing the robot to determine the rough location of the instructor’s hand. This way a starting signal and a region of interest are given for the following processes.

The hand tracking is based upon a saliency map composed of different visual cues, such as change in gray value, color, and stereo disparity. A hand moving in the plane of fixation of the camera system shows up as a blob of high intensity in the saliency map, which is sufficient for localization. A detailed account of these cue fusion techniques employed to establish the salience map can be found in (Triesch, 1999).

After the initial hand detection the fingertips are located to serve the robot as a cue on how to orientate its endeffector. In order to represent the finger tips so called compound jets are employed. A compound jet differs from an ordinary jet, as described in section 2.2.2, as its components are not exclusively derived from the Gabor transform of the gray value image, but also from the Gabor transform of, e.g., the skin color similarity image.
4.2. FINGER TRACKING FOR IMITATION LEARNING

Using this model and some slight geometrical constraints the fingertips are found in the 128 × 128 pixels region of interest determined by the hand tracking (see again (Triesch, 1999) for details).

Fingertip Tracking

Following the detection stage the teacher’s fingertips are tracked through a continuous sequence of stereo images while he performs the grasping movement. To this end phase-based tracking as described in section 3.3 is again used. But in order to meet the challenges of the given circumstances the phase-based tracking is augmented to also incorporate different visual cues beyond the gray value images used so far and to deal with the geometrical constraints given by the stereo camera setup (see below for a detailed discussion).

Imitation

The robot stops the tracking of the instructor’s fingertips when the trajectory shows a large displacement perpendicular to the table, which results from the object pick-up and placement on the table.

Thereafter the observed movement is imitated. To this end the trajectory in space is estimated from the trajectory measured in the stereo images. The robot then move his endeffector to a position on the trajectory 5cm from the object and performs a movement based on a linear approximation of the observed movement (for details see (Maël, 1996; Triesch et al., 1999)). Figures 4.3 to 4.5 show the different stages of a successful imitation.

4.2.2 Phase-Based Tracking of Fingertips

The tracking of fingertips through a sequence of stereo image pairs is in principle a straightforward application of the previously discussed point tracking scheme based on Gabor jets (see section 3.3), but matters are complicated by the fact that fingers in contrast to, e.g., faces only have very little internal structure. As a result points on the fingertip are fairly ill-defined and thus hard to track. The border between the fingertip and the background on the other hand is well defined. But tracking this border region is a process which is very susceptible to influences from the background. Therefore two modifications to the single point tracking are made, which increase robustness and reliability.

Tracking with Compound Jets

Compound jets as introduced in (Triesch and v.d. Malsburg, 1996) provide a way of integrating multiple visual cues, such as motion, color or gray value, into a local point description as required by elastic bunch graph matching. By taking more information in account than the ordinary jets they significantly reduce ambiguity, as was shown in the
context of object recognition (see (Triesch and v.d. Malsburg, 1996; Triesch and Eckes, 1998; Triesch et al., 1999; Triesch, 1999)).

To make use of additional cues in the phase-based tracking, it is once again possible to estimate the disparities independently of each other for all cues and obtain a resulting disparity by a suitably weighted average of the individual results. But as shown before in the context of geometrical constraints it is advantageous to aim for integration early in the process of disparity estimation. As the disparity \( d \) at a given point should be the same for all the different cues, compound jets can easily be integrated into the phase-based tracking by adding only a further summation to equation 2.10 over all cues with suitable weighting, such that

\[
\begin{align*}
    s_{\text{disp}} &= \frac{\sum_i w_i \sum_k a_{ik}^j(x)a_{ik}^j(x') (1 - 0.5(\phi_k^1(x) - \phi_k^2(x') - \hat{k}^T \tilde{d}))^2}{\sqrt{\sum_i w_i \sum_k a_{ik}^j(x)^2} \sqrt{\sum_i w_i \sum_k a_{ik}^j(x')^2}},
\end{align*}
\]

where the different cues are indexed by \( i \). This way different visual cues are integrated into the phase-based tracking very much in the same spirit as the geometrical model assumptions were integrated via internal constraints (see section 3.3.2).

In the present context of tracking fingertips through a grasping movement the cues chosen are jets derived from gray value images (\( I \)), skin color similarity images (\( S \)) and from the associated difference images (\( \Delta I, \Delta S \)), which were all subjected to Gabor transformations with standard parameters to derive the compound jets. The weights \( w_i \) were chosen \( w_I = w_S = \frac{1}{6} \) and \( w_{\Delta I} = w_{\Delta S} = \frac{2}{6} \).

**Stereo Constraints**

In the previous chapter is was shown that geometrical constraints can significantly improve tracking. Although constraints, as they were used for tracking head movements, cannot be imposed on the movement of fingertips due to the high number of degrees of freedom of the human hand, the fact can be exploited that the description of a moving point as given by two stereo images is redundant.

The location of a point situated in the physical world is accurately described by three coordinates in some suitably chosen Euclidean space. But the projection of this point onto the image planes of a stereo camera system yields a four-dimensional vector, namely the \( x \) and \( y \) position of the point in the image plane of each camera. The four coordinates, as measured by the camera system, can therefore be parameterized in terms of three hidden variables, being the point’s coordinates in the three-dimensional physical space. They are hidden in the sense that they are not visually accessible without further model assumptions.

To see whether a suitable model can be derived exploiting the previously introduced techniques it is instructive to look at the relation between the observed projected coordinates \( x_l, y_l, x_r, y_r \) and the hidden real world coordinates \( c_x, c_y, c_z \) of a given point, assuming
that the stereo camera setup is well described by a fairly simple geometrical model as discussed in (Pagel, 1997; Pagel et al., 1998). For a simple camera system with a maximal view angle $\Phi^{max}$, a pixel resolution $r$ in $x$ and $y$ direction, a vergence $v$, and distance $s$ between the two cameras the relation is then given by

$$\begin{align*}
x_l &= \frac{r}{2 \tan(\Phi^{max})} \frac{\cos\left(\frac{\Phi}{2}\right) \left( c_x - \frac{s}{2} \right) - \sin\left(\frac{\Phi}{2}\right) c_z}{- \sin\left(\frac{\Phi}{2}\right) \left( c_x - \frac{s}{2} \right) - \cos\left(\frac{\Phi}{2}\right) c_z} \\
y_l &= \frac{r}{2 \tan(\Phi^{max})} \frac{- \sin\left(\frac{\Phi}{2}\right) \left( c_x - \frac{s}{2} \right) - \cos\left(\frac{\Phi}{2}\right) c_z}{\cos\left(\frac{\Phi}{2}\right) \left( c_x - \frac{s}{2} \right) + \sin\left(\frac{\Phi}{2}\right) c_z} \\
x_r &= \frac{r}{2 \tan(\Phi^{max})} \frac{- \sin\left(\frac{\Phi}{2}\right) \left( c_x + \frac{s}{2} \right) + \cos\left(\frac{\Phi}{2}\right) c_z}{\sin\left(\frac{\Phi}{2}\right) \left( c_x + \frac{s}{2} \right) - \cos\left(\frac{\Phi}{2}\right) c_z} \\
y_r &= \frac{r}{2 \tan(\Phi^{max})} \frac{\cos\left(\frac{\Phi}{2}\right) \left( c_x + \frac{s}{2} \right) - \sin\left(\frac{\Phi}{2}\right) c_z}{- \sin\left(\frac{\Phi}{2}\right) \left( c_x + \frac{s}{2} \right) + \cos\left(\frac{\Phi}{2}\right) c_z}.
\end{align*}$$

The projected coordinates are obviously not confined to a linear subspace by the above constraints, but rather to a non-linear manifold as visualized in figure 4.2. In section 3.3.2 it was shown that non-linear constraints can be dealt with in terms of tracking by local linearization of the constraining manifold.

Deriving the constraining manifold, on the other hand, by the strictly linear techniques developed in chapter 3 is not possible. For the appearance manifold of objects the problem of mapping non-linear manifolds will be discussed in the following chapters and it will be shown that locally linear approximation can also be employed to extract the overall transformation properties.

In the here specified task such a very general method of deriving the constraining manifold is not required, because a fairly accurate geometrical model can be easily established (see above) and the necessary refinements can be obtained by exploiting the robot’s ability to interact with the environment. To this end the robot observes the projected image of a LED with its camera system, while changing and measuring the light bulbs position in space using his end effectors as described in (Pagel, 1997; Pagel et al., 1998). This way the coordinates as observed by the camera system and the physical coordinates as given by the position of the end effectors of any given point can be measured and used to refine the initial geometric model.

The such derived model can now be used to perform the tracking directly in terms of the physical point coordinates using the local linearization of the manifold as described in section 3.3.2. This has the advantage of eliminating one degree of freedom for every tracked point and of directly associating the tracked points with physical coordinates, thus facilitating the interpretation of the observed motion in terms of properties of the physical world.
Figure 4.2: **Stereo Manifold:** A projection of the three dimensional manifold in the four dimensional space spanned by the x and y position of a point in the left and right camera plane is shown. The projection was derived by assuming that the y coordinate of the point is fixed for the right camera resulting in the two dimensional manifold in the three dimensional space shown here. The assumed camera parameters were $v = 50^\circ$, $\Phi^{max} = 45^\circ$ and $r = 100$ pixel.

4.2.3 Discussion

The system’s performance varied most strongly with the complexity of the background in front of which the teachers movements were to be observed. In front of completely homogeneous background the fingertips were reliably located and tracked through the movement. The grip was then imitated by the robot without problems. The deviation of the trajectory performed by the robot as compared to the teacher’s was around 1cm. Even in front of mildly textured background as the one shown in figure 4.5 accuracy was still satisfactory. If the background complexity was further increased such that strong discontinuities in the gray value as well as in the color components were introduced fingertip detection and tracking started to fail.

Unfortunately it was not possible to perform systematic and quantitative tests of the system’s performance due to endless difficulties with the robot’s hardware.

Despite that, we believe to have shown that the tracking techniques developed in
section 3.3.2 can be flexibly employed to realize a tracking built upon cue fusion, as advocated in (Triesch, 1999), and to integrate non-linear constraints to reduce ambiguities. On the other hand it became clear that the way of deriving transformation properties via principal component analysis as laid out in chapter 3 is not sufficient to deal with non-linear constraining manifolds. The problem of how to deal with those non-linearities if the constraining manifold can only be derived from passive observation and not as in the above case from active manipulation of the environment is tackled in the following chapters.
Figure 4.3: **Hand and Finger Detection:** After the hand is located in the two stereo images (indicated by the large circles), the fingertips are detected by compounds jets. The fingertips are marked by smaller circles and the approximate pointing direction is indicated by a line.

Figure 4.4: **Fingertip Tracking:** After the detection phase the fingertips are tracked through a continuous sequence of stereo images. Displayed is the last frame of this sequence for the left and right camera image. The tracked trajectory for both fingertips are shown by the superimposed black line. The final estimated fingertip positions are depicted by white circles.
4.2. FINGER TRACKING FOR ImitATION LEARNING

Figure 4.5: **Imitation:** Having observed the grip movement performed by the teacher the robot assumes the starting position 5 cm from the object on the trajectory (see figure 4.5(a)) and grasps the object 4.5(b) following a linear approximation to the observed trajectory.
Chapter 5

Representation by Perceptual Manifolds

In the previous two chapters it was shown that given a constraining subspace, be it linear like in chapter 3 or non-linear as in section 4.2, significant improvements are possible through the enhanced level of object knowledge contained in the system. In chapter 3 it was demonstrated how a linear subspace of typical object transformations can be derived from very little a priori knowledge in the case of frontal faces and how those transformation properties yield an understanding of the three-dimensional structure of faces. But the approach was fundamentally limited in three ways.

- It required an initial representation of faces, provided by the bunch graph.
- Only linear representations of the transformation properties could be derived.
- The system had no way of determining whether a linear representation was sufficient such that the training database had to be chosen beforehand to limit the extend of the contained transformations.

The goal of the following chapters is to overcome these limitations and to create a complete object description from a limited number of single views. In this chapter first some general remarks concerning the representation of three-dimensional objects are made and the approach pursued in the following chapters is laid out.

5.1 Representational Spaces

As pointed out in chapter 1, one goal of an object representation must be to establish a second order isomorphism. This means that the representation must establish relations between the perceived views which reflect the physical properties of the object. If this is achieved, knowledge acquired for a limited number of views can be generalized by inter-
and extrapolation as to be applicable for all possible views. If, e.g., a robot has grasped an object successfully from a number of orientations and the object is then presented in a new orientation whose relation to previously experienced orientations can be established by virtue of the object representation, the correct grasping movement might be determined from interpolating the earlier performed movements. It is thus our goal to retrieve an object representation which facilitates this kind of generalization or learning from few views to all possible views. In contrast to (Baker and Nayar, 1996; Mel, 1997; Nelson and Selinger, 1998) our main emphasis is therefore not on designing an object representation maximally efficient in terms of storage capacity or recognition speed, but rather on an object representation which accurately portrays the physical properties of the object.

5.1.1 Physical Object Space

One straightforward object representation which exhibits the desired qualities is given by the one provided by geometrical considerations. An object can simply be described by a set of points in a three-dimensional Euclidean space. The state of any rigid object in this representation is uniquely defined by a set of six variables, describing translation and rotation within the given frame of reference. Smooth invertible mappings from this six dimensional parameter space onto the state space describing the current locations of the object’s points in Euclidean space can easily be derived. All object orientations can thus be parameterized and a metric of the orientations is given by the metric on the parameter space.

Because of this metric defined by the parameterization, this object representation is well suited to guide interactions with objects such as grasping, catching, obstacle avoidance and the like. For this reason computer vision has long aimed at retrieving such a representation directly from the visual input. This approach most prominently features in the so called shape from X paradigm, where different visual cues are exploited to solve the inverse projection problem in order to obtain coordinates in a three-dimensional Euclidean space for distinct points on an object’s surface. Unfortunately, this has turned out to be a very unstable and unreliable process.

5.1.2 Appearance Object Space

A different approach to representation, which is less challenging from a computer vision point of view, is centered around the notion of describing an object and its relation to an observer solely in terms of its visual appearance. This is, of course, a very general idea. In order to allow an instructive discussion some additional assumptions must be made. So let us assume that a recognition system is given which is only sensitive to an object’s identity and its physical relation towards the observer, ignoring, e.g., illumination issues and camera characteristics. Let us further assume that slight changes in observer-object-relations result in slight changes in visual appearance such that the mapping from the
5.1. REPRESENTATIONAL SPACES

physical states to the appearances is smooth. Elastic graph matching can be viewed as a crude approximation to such a recognition system. If the visual appearance can now be associated with the same set of six variables that uniquely determine the physical state of the object, this direct relation between a known object’s appearance and its physical state would allow easy interaction with the object based upon visual perception. However, fundamental as well as practical matters complicate the issue considerably.

Even if a parameterization is found that associates a given set of parameter values uniquely with a certain visual appearance, the inverse is generally not the case. Three cases can be distinguished.

1. **Fully Invertible Parameter Mapping:** In the simplest of all cases a given view of an object uniquely determines its physical parameters (see figure 5.1). Given the smoothness assumptions stated above, the views of an object constitute a manifold in appearance space and this manifold can be charted by the physical parameters. Via those physical parameters a metric would be introduced on the views, which is identical to the one given by geometrical considerations realizing all the advantages. Moreover, any set of charts to the appearance manifold which is retrieved can be mapped to those charts given by the physical variables via a reparameterization. As such it is smooth and topology preserving, which greatly facilitates the association of the views with the physical parameters. Putting it differently, the learning of the relations between views and physical parameters or any other property related to them is facilitated by retrieving the view manifold beforehand.

2. **Invertible Parameter Mapping on a Subset of the Parameter Space:** For symmetric objects, such as a cube or a cylinder without surface markings, a given view does not uniquely determine the physical parameters. The symmetry introduces ambiguities. But given the smoothness assumption their views still constitute manifolds in appearance space. Although the mapping from the physical parameter space onto the appearance space is not invertible, subsets of the physical parameter space can be found such that the mapping is invertible and the inverse is smooth for all views. A smooth topology-preserving reparameterization in terms of physical parameters is thus still possible.

3. **Partially Invertible Parameter Mapping:** The situation is very different for objects with local symmetries, e.g., a cube where two faces are identical and all others differ. Due to self-occlusion the structure in appearance space is self-intersecting. This in turn means that it is not a manifold and the inverse mapping from the views onto the physical parameters cannot be smooth around those points of intersection. One could say that at those points of intersection virtual transitions are introduced, because views with are perceived to be neighbors are not neighboring in terms of their physical parameters.
An object might even exhibit different degrees of freedom in appearance space depending on the current point of view, as for example a sphere with a small surface marking. Whenever the surface marking is visible the structure in appearance space has six degrees of freedom, but if it is occluded only three remain.

In those cases only subsets of all possible views can be represented in terms of manifolds. In the above two cases this means splitting the views into two subsets which have to be treated separately, and excluding the points with virtual transitions, which can not be described in terms of a manifold.

These problems in retrieving a mapping from the appearance space onto the space of physical parameters are exacerbated by the need to recognize views in the presence of
unknown transformations and to learn from a limited number of views. As discussed in chapter 1 recognition of unknown views requires some amount of generalization, i.e. views must be perceived as being the same even if they differ slightly. If a view representation is not chosen well it either fails to recognize views as identical or identifies unrelated views with one another, thus creating situations where the topology of views is not represented properly. When retrieving the manifold in appearance space from a limited number of views the same problem occurs if the neighborhood of views is used to determine whether they are neighbors in terms of physical parameters.

In relation to the problem of creating an object representation solely from visual appearance two things must be concluded from the above discussion. Firstly, choosing the right measure of similarity between views is as crucial as in discrimination tasks, such as, e.g., face recognition. And secondly, because the overall topology of views in appearance space is fairly susceptible to local distortions, such as at points of virtual transitions, the need to establish an accurate local topology which allows to confine the influence of those distortions must be acknowledged.

In the following a short overview is given on how we plan to retrieve the topological structure of views.

5.2 Retrieving the Topology of Views

The goal of the following two chapters is to retrieve the structure of an unknown object’s appearance space from a limited number of single views, in a way that on the one hand allows to apply the principles developed in the previous chapters and on the other hand facilitates the learning of mappings between views and parameters related to the state of the object in the physical world.

To this end we assume that a subset $U$ of views of an object’s viewsphere, i.e. the set of all possible views, is given. No restrictions are imposed on $U$ in terms of object-observer-relations, contrary to the database used to derive the transformation properties of faces, where only frontal views of faces were included. In order to apply the principles derived in chapter 3 two things must be accomplished. Firstly the views in the training dataset must be represented in terms of graphs and secondly subsets of views must be retrieved whose transformation properties are well described by the linear approximations given by principal component analysis.

5.2.1 Aspect Representations

To arrive at a view description in terms of graphs the spin segmentation approach developed in (Vorbrüggen, 1995; Eckes and Vorbrüggen, 1996) is employed. This approach allows us to separate the object shown in an image from the background. From the thus determined object region a jet-labeled grid graph can be extracted (see section 6.1.2).
A graph of a single view, which was derived this way, can now assume the role of the bunch graph of chapter 3. But because the training dataset is supposed to contain large variations in viewing angle, which can certainly not be approximated linearly, a subset must be chosen which is suitable to derive the linear approximations to the graph’s transformation properties. To select suitable views a graph similarity is used very similar to the one employed in the face recognition experiments in section 3.2. The graph is matched onto all views in the training dataset and whenever the resulting similarity is above a threshold value $t$ the view is merged into the subset. In this context we call the graph a representational graph, and adopting the terminology of (Koenderink and van Doorn, 1979), the subset of views it defines is called an aspect of the object. A representational graph and its associated aspect can now be treated exactly as the bunch graph and the gallery of frontal faces in section 3.1.2. This means the correspondence between the representational graph and all views from its aspect is established via elastic graph matching. Principal component analysis can then be performed on the resulting correspondence mappings. This way a representation of the aspect is obtained.

From the eigenvalue distribution it is then possible to derive an estimate of the number of degrees of freedom within an aspect, which in turn is also an estimate of the dimensionality $K$ of the view manifold the aspect is a part of. Knowing the dimensionality $K$ a topology can be established on the views of the aspect, by assigning to the correspondence mappings associated with each view the coordinates in the coordinate system spanned by the first $K$ principal components (see section 6.2.1).

This way it is possible not only to derive the transformation properties of the representational graph but also to establish a topology on all views in the aspect it defines.

5.2.2 Manifold Representation

After having established topologies and parameterizations of views within the individual aspects via principal component analysis, the goal must now be to integrate those local descriptions into a single global description. This global parameterization would then establish a topology of the entire view manifold, thus providing a complete understanding of the objects transformation properties. To this end we seek to cover the whole view manifold with local aspects and then derive a global embedding parameter space, into which the aspects’ local parameter spaces can all be mapped.

In theory every view within the training dataset can be used to derive a representational graph and to define an aspect. For practical purposes it is sufficient to pick only a few views from which to generate the representational graphs and the associated aspects, such that all views within the training dataset are part of at least one aspect and aspects which represent neighboring areas on the view manifold have some views in common. This way the view manifold is completely covered by the local topologies generated by the aspect representations and the views contained in more than one aspect can be used to establish relations between aspects.
5.2. RETRIEVING THE TOPOLOGY OF VIEWS

To retrieve a global embedding parameter space, distances between the aspects are estimated by exploiting those views which are part of more than one aspect. Via *metric multidimensional scaling* (see section B of the appendix) those distances are converted into coordinates in an embedding parameter space for all aspects. By calculating linear mappings from the individual aspects into the embedding parameter space all views acquire coordinates in this common frame of reference. In this fashion a global topology is established for all views.

In order to demonstrate how this topological ordering facilitates learning of relations between physical variables describing the state of the object and the perceived views, a linear interpolation scheme on the retrieved manifold is set up. Utilizing this interpolation scheme a mapping between the views and the physical viewing angles is obtained from very few examples.
Chapter 6

Aspects

This chapter deals with the problem of applying the principles of chapter 3 if no initial object representation, such as a bunch graph, is attainable and the given training dataset is composed of views which are not controlled in terms of viewing angles. To this end it will be first discussed how view representation in terms of a model graph are obtained from single images and later how subsets are retrieved from the training data which are suitable for the linear approximation techniques developed in chapter 3.

6.1 View Representation by Labeled Graphs

No initial object representation is available now on which to base the whole treatment of transformation properties as in chapter 3. With the help of a bunch graph as pre-existing object representation a whole class of objects, namely faces, could be handled within a single consistent representational framework. Here, an appearance-based object representation must be derived from nothing but the visual input. This immediately raises the question of how to determine the location of the object of interest within a given image.

6.1.1 Segmentation

Dividing an image into meaningful regions without any information on what the image might be showing requires data-driven segmentation. Data-driven segmentation is a very complex and important subject on its own, with many unsolved fundamental problems. A detailed discussion of this issue is beyond the scope of this work, but can for example be found in (Vorbrüggen, 1995). To facilitate this general problem in the course of this work some restrictions are imposed on the admissible image data to assure that segmentation can reliably be achieved with relatively little effort. Firstly, it is assumed that images presented to the system during the learning phase contain only one object. Secondly, the background is required to be fairly homogeneous. And lastly it is assumed that the object
Figure 6.1: **Single View Models**: In order to describe an object, a suitable representation of a single view is found by first splitting the image into a background and a foreground region. To this end the spin segmentation developed by Jan Vorbrüggen, Christian Eckes and Christoph von der Malsburg (Vorbrüggen, 1995; Eckes and Vorbrüggen, 1996) is used, in connection with the knowledge that only one object is presented approximately at the center of the image and that there is a nearly homogeneous background. After the segmentation step the resulting object region is represented by a jet-labeled graph.

is more or less in the center of the image. Given those restrictions the image is processed the following way.

The image is decomposed into areas of similar gray value, using the spin segmentation approach described in (Vorbrüggen, 1995; Eckes and Vorbrüggen, 1996). To each pixel a label is assigned identifying the area it belongs to. For each area the center of gravity in the image is calculated as well as the pixels’ mean distance from it. Using the assumption of a single object centered in the image in front of an approximately homogeneous background, the area with the largest mean distance is assumed to be the background. All pixels outside this region are now considered non-background pixels. The largest connected region of those non-background pixels is then assumed to constitute the object (see figure 6.1).

### 6.1.2 Extracting and Matching Graphs

Having segmented the image into two regions, one containing the background and one containing the object, a representation of the object view can be created. As usual, graphs labeled with Gabor jets are used to yield an initial view-based model. A grid graph is created by placing nodes 8 pixel apart on the region containing the object. Neighboring nodes are connected by an edge. Each node is labeled by a jet, which serves as a local image description. For the associated Gabor transform standard parameter settings are used.

As demonstrated before, this kind of representation is well suited to recover similar views of the same object from image data via *elastic graph matching*. Therefore a matching schedule is adopted, which is almost identical to the one employed in section 3.1.1. Translation is again compensated for explicitly by using a *scan translation move* with the $s_{abs}$ similarity function. All remaining transformations are treated by a *scan local move*.
6.2. FROM VIEWS TO ASPECTS

with $s_\Phi$, $d_{\text{local}}$, which determines the amount of local deformation (see section 2.3.2), is set to $d_{\text{local}} = 10$.

Because elastic graph matching will later be used to judge whether views are neighboring on the view manifold in appearance space and, as pointed out in chapter 5, this is a crucial and potentially very sensitive step when retrieving the view manifold, the similarity measure which estimates the similarity between a view and a graph after the correspondences are established is slightly modified. Instead of taking the mean similarity over all nodes, the total similarity is given by the minimum similarity over all nodes:

$$s_{\text{average}} = \frac{1}{N} \sum_{i=1}^{N} s_i$$ (6.1)

(see section 2.3.3) is replaced by

$$s_{\text{min}} = \min_{i} (s_i).$$ (6.2)

This way we hope to exclude corrupt or misleading data, which otherwise might seriously impair the derived object representation.

6.2 From Views to Aspects

Given a model graph $g$ and a set of views $U$, the aspect $a$ is defined as the subset of views $u_i$ of $U$, whose similarity $s_{\text{min}}$ to $g$ is above a given threshold $t$, i.e.

$$a = \{ u_i \in U | s_{\text{min}} (g, u_i) > t \}. \quad (6.3)$$

In this context we call $g$ the representational graph to the aspect $a$.

It was shown in (Kefalea, 1998; Becker et al., 1999) that graph similarities can be used successfully to determine the orientation of an object via a nearest neighbor approach. One can therefore entertain the hope that the similarities reflect the neighborhood relations between views accurately enough such that a threshold $t$ can be found which allows to limit the amount of variation in terms of viewing angles within each aspect. If this is indeed the case, the model graph $g$ can assume the role of the bunch graph in chapter 3 and a linear model of his transformation properties suffices to correctly represent the correspondence mappings between $g$ and the views in the aspect $a$. An aspect is thus assumed to be a subset of $U$ which can successfully be dealt with by employing the linear techniques derived in chapter 3.

Before the issue of choosing an appropriate value for the threshold $t$ is discussed, we will explain how the transformation properties of a representational graph are exactly derived and how the transformation properties in combination with the representational graph yield a representation of an aspect.
6.2.1 Local Linear Descriptions

Let $g$ be a representational graph and $a$ its associated aspect, the correspondences between $g$ and all views in $a$ are then also given, as $a$ was derived from the training set $U$ by thresholding on the graph similarities, which in turn were established by elastic graph matching using the above described matching schedule.

As in chapter 3 a correspondence mapping can again be associated with a point $\mathbf{X}_i = (x_1, y_1, \ldots, x_N, y_N)$ in Euclidean space, where $(x_l, y_l)^T$ is the mapped position of the $l$-th node of the representational graph. At this point it must be noted that for a given aspect the dimensionality of the mapping space is fixed and given by twice the number of nodes $N$ of the representational graph, but between different aspects associated with different representational graphs the dimensionality might vary.

The correspondence mappings are now represented in the same way as those in chapter 3, and one could proceed in the same fashion. But as translation invariance can easily be achieved by ordinary graph matching and as its properties are straightforward, there is no need to derive it from the perceived data for each object individually, especially as its feasibility was already demonstrated in chapter 3. The data variation due to translation is thus explicitly removed by subtracting the center of gravity from each mapped node position, so that a correspondence mapping is actually represented by

$$\mathbf{X}_i = \left( x_1 - \frac{1}{N} \sum_n x_n, y_1 - \frac{1}{N} \sum_n y_n, \ldots, x_N - \frac{1}{N} \sum_n x_n, y_N - \frac{1}{N} \sum_n y_n \right).$$

(6.4)

It must be noted that rotation in plane is just like translation a transformation which can be applied directly to the views. Therefore, observed rotation in plane does not reveal any information about an object not contained in the view being rotated. Consequently, in-plane rotation could be dealt with in the same way as translation. We here explicitly choose not to do so, because it would increase the matching complexity considerably and also because the psychophysical data seems to suggest that invariance towards in-plane rotation in the human visual system is not as developed as translation invariance (Farrell and Shepard, 1981). Thus in-plane rotated views will not be identified with each other later on.

Principal component analysis is then performed on the set of all correspondence mappings $\mathbf{X}_i$ associated with the views in the aspect. The resulting eigenvector matrix $P$ and the mean $\bar{M}$ then provide the linear approximation of the transformation properties of the representational graph $g$ (see section 3.1.2). This way the information, which was successfully exploited in chapter 3 to constrain the correspondence problem in a number of scenarios, can be derived for subsets of $U$ without reference to an initial object representation.

But the representational graph $g$ and the eigenvector matrix $P$ can also be used to introduce a topological structure on the aspect. For a given dimensionality $K$ of the aspect, i.e. the number of degrees of freedom along which the observer-object-relations
varied to create those views, the first $K$ eigenvectors $\bar{P}_1, \ldots, \bar{P}_K$ form an orthogonal basis, which spans an affine subspace of the correspondence mapping space. A correspondence mapping $\bar{X}$ can then be expressed in this coordinate system by the vector

$$\bar{\xi} = \left( (\bar{X} - \bar{M})^T \bar{P}_1, \ldots, (\bar{X} - \bar{M})^T \bar{P}_K \right)^T.$$ 

(6.5)

As each view within the aspect is associated with a correspondence mapping, these coordinates impose a topology upon the views of the aspect. Moreover the representational graph $g$ contains also the local image descriptions provided by the jets and for each view $i$ in the aspect the coordinates $\bar{\xi}_i$ yield the approximate correspondence between $g$ and the view, such that an approximation to the view $i$ could be reconstructed from this representation, similar to the reconstructions derived in section 3.1.3.

The quality of the above representation depends, of course, critically on the ability to retrieve aspects from the training dataset $U$ which are well represented by a $K$ dimensional affine subspace, and the ability to estimate the number of degrees of freedom $K$ within an aspect. To this end a suitable value for the threshold $t$ needs to be found and a method to determine the correct dimensionality $K$. In order to judge the influence of $t$ on the quality of the aspect representation two key properties of the aspect representation are of importance.

- **Reconstruction Error:** A view whose correspondence to the representational graph is given by $\bar{X}$ is represented in terms of a $K$-dimensional linear model as

$$\bar{X}_{rep} = \bar{M} + \sum_{k=1}^{K} \left( (\bar{X} - \bar{M})^T \bar{P}_k \right) \bar{P}_k.$$ 

(6.6)

The $\bar{P}_k$ are the sorted eigenvectors as introduced in section 3.1.2 and $\bar{M}$ is the mean correspondence mapping within the given aspect. The reconstruction error is now given by $\| \bar{X} - \bar{X}_{rep} \|$. It must be noted that this error measure is only concerned with the arrangement of local features not with the transformations of the local features themselves. But by virtue of the construction and the invariance properties of the features (see section 2.2.3) this difference is assumed to be small. For a detailed discussion of the transformation properties of jets see (Okada et al., 2000; Okada, 2001).

- **Preservation of Topology:** In order to go beyond the representation of the training data more than faithful reconstruction is required. The topology, which is established on the views by representing them in a coordinate system spanned by the $\bar{P}_k$, must reflect the neighborhood relations as determined by the associated physical parameters. The topology of views should thus be the same in terms of a real world parameterization and in terms of a parameterization in principal components.

Whereas the the reconstruction error is easily estimated from the training dataset, the preservation of topology is not quantified easily.
6.2.2 Estimation of Topology Preservation

The usual way of estimating how well a certain mapping preserves neighborhood relations, is by displaying examples in an intuitive fashion. Figure 6.2 shows some examples of aspect representations.
representations for given thresholds $t$. The views are plotted in the two dimensional coordinate system spanned by the first two principal components, i.e. view $i$ is plotted at position
\[
\tilde{\xi}_i = \left( (\tilde{X}_i - \tilde{M})^T \tilde{P}_1, (\tilde{X}_i - \tilde{M})^T \tilde{P}_2 \right)^T .
\] (6.7)
From independently available information on the angles under which the images were taken the neighborhood relations on the view sphere are taken and visualized by connecting those views with each other which are neighbors in terms of viewing angles.

In order to evaluate the quality of representation a quantitative measure of topology preservation is needed. A suitable measure is given by the so called topographic product, which was proposed in (Bauer and Pawelzik, 1992). It was developed to estimate the dimensionality mismatch of the input and output space of selforganizing maps. The measure is based on the simple notion that for any given data point before and after the mapping the $r$ nearest neighbors should be the same, for any value of $r$. By weighing violations appropriately the following formulation is derived.

The topographic product can be estimated for a mapping $h$ from a metric space $V$ onto a metric space $W$ and a given set of $M$ data points $\{v_1, \ldots, v_M\}$ and the corresponding $\{w_1, \ldots, w_M\}$ in the output space, such that $w_j = h(v_j)$ The distance measures in $V$ and $W$ are denoted $D^V$ and $D^W$ respectively. The index of $r$-th nearest neighbor of $v_j$ is given by $n^V_r(j)$ and the index of $r$-th nearest neighbor of $w_j$ by $n^W_r(j)$. In this case the topographic product $T$ is given by
\[
T = \frac{1}{M(M-1)} \sum_{j=1}^{M} \sum_{r=1}^{M-1} \left| \ln \left( \prod_{l=1}^{r} \frac{D^V(v_j, v_{n^V_l(j)})}{D^W(w_j, w_{n^W_l(j)})} \right) \right| .
\] (6.8)
As the original topographic product $T$ was derived in order to detect systematic defects in the mapping, due to mismatches in the chosen dimensionalities of input and output space, the formula had to be slightly modified by taking the absolute value of the logarithm to also yield sensible results for randomly distorted mappings.

In our case, $h$ is the mapping from the ground truth viewing angles to the coordinates $\tilde{\xi}_i$ of an aspect representation. $D^V$ is there the ordinary metric between angles and $D^W$ the Euclidean metric in the aspect representation space.

### 6.2.3 Choosing a Threshold

In order to estimate the quality of aspect representations and to choose a sensible similarity threshold $t$, aspects were extracted for varying values of $t$ from databases of different objects with angle variations between neighboring views between $3.6^\circ$ and $7^\circ$. A detailed account of the database from which these samples were drawn can be found later on in section 7.1.5. Figure 6.3 shows the mean aspect size, the mean reconstruction error, and...
the mean topographic product as a function of $t$. The monotonous decline in reconstruction error with an increase in threshold value is consistent with the assumption that the transformation properties of views are inherently non-linear and that the smaller the aspects are constructed the better the linear approximation. The topographic product on the other hand seems to converge towards a baseline for increasing values of $t$. At a point of approximately $t = 0.7$ topology preservation is achieved as far as possible. Given the apparent impossibility to satisfy the requirement of a good linear approximation and large aspects at the same time, the threshold is set to $t = 0.7$ for all conducted experiments, as the reconstruction error is small at that point and the topology of views is reflected as well as possible.

### 6.2.4 Estimating Local Dimensionality

In the previous experiments the dimensionality of the linear aspect models was manually set to the number of degrees of freedom contained in the data. To automatically estimate the number of the object’s local degrees of freedom, which determines the dimensionality, a modified version of the Scree-test is used (Fahrmeir and Hamerle, 1984). Given the eigenvalues of the principal components of a distribution, it is assumed that they can be separated into two classes, one belonging to variations due to changes in some underlying variables and the second being created by noise. The first group, consisting of the largest eigenvalues, contributes most to the overall variation. The second group, generated by noise, usually shows a characteristic flat and monotonous decline, which is approximately exponential (Cattel, 1966).

To detect this transition the ratio between the eigenvalue $p_i$ and its immediate pre-
6.3 Discussion

In this chapter it was demonstrated how to overcome two of the major drawbacks of the approach outlined in chapter 3. Firstly, it was shown how an initial object representation can be derived directly from single images, via data driven segmentation.

Secondly, it was shown that the view representation by labeled graphs can be exploited to extract aspects from an unconstrained set of training views, such that the principles

Figure 6.4: Estimating Dimensionality from Eigenvalues: (a) shows the eigenvalues in descending order for an example representation with two degrees of freedom. (b) shows the corresponding ratios of \( r_i = \frac{p_i}{p_{i-1}} \), with \( r_1 \) set to 1. A clear minimum can be seen for \( i = 3 \).

decessor \( p_{i-1} \) is calculated. The minimum of those ratios \( \frac{p_i}{p_{i-1}} \) is supposed to mark the transition. So the first \( K \) principal components are assumed to be due to variations in the object’s degrees of freedom if

\[
K = \arg \min_i \left( \frac{p_i}{p_{i-1}} \right) - 1. \tag{6.9}
\]

An example distribution is shown in figure 6.4. Using this method, a correct estimation of the number of underlying degrees of freedom was possible for 97% of the aspects.

If more than one aspect is known to belong to a single object and those aspects are known to share the same degrees of freedom, which is not true in general but holds in the simple cases discussed below, the classification accuracy can be increased by subjecting not the eigenvalues of aspect separately to the Scree-test, but rather the mean eigenvalues of all aspects. In all experiments performed this modification led to an accurate estimation.
of chapter 3 can successfully be employed to derive the transformation properties of the view representations. Moreover, a representation was established on the aspects, which on the one hand imposed a topological ordering on the views contained in an aspect and on the other hand could determine the actual dimensionality of an aspect.

To create a representation of an entire viewsphere from a training set $U$, aspects can now be derived from $U$ such that each view is at least part of one such aspect. This way local topologies are introduced for all views. The topologies are local, because so far the single aspects are completely unrelated, i.e. no neighborhood relation is established between the views of different aspects.

Via elastic graph matching and subsequent projection of the correspondence mapping onto the aspect representation coordinates in a Euclidean space can be assigned to each view. In turn, to each point in the Euclidean space of an aspect representation a view representation can be created from the aspect’s representational graph (see section 6.2.1). In this sense the aspects introduce invertible mappings from the appearance space of an object onto an Euclidean space. The aspects can therefore be said to constitute charts to the view manifold of an object.

In order to derive an atlas of charts, which parameterizes the entire manifold, mappings between the single charts still need to be established.
Chapter 7

View Manifolds

In this chapter an approach is described to integrate the single local topologies of the aspect representations into one global topology for all views. To achieve this, relations between the individual aspects are established yielding a complete parameterization of the view manifold.

As a final step it will be shown how to reap the benefits of the resulting object representation. To this end the views are related to parameters describing physical object properties, namely the viewing angles.

7.1 From Charts to Manifolds

In order to arrive at a complete parameterization of the view manifold it is assumed that a training set $U$ of views is given. It is also assumed that a set of representational graphs and the aspects they define were extracted from $U$ such that all views in $U$ are part of at least one aspect and that pairs of aspects which are neighbors in appearance space have a set of views in common. The set of views common to two aspects $a$ and $b$ is denoted $O_{ab}$. Because these sets $O$ contain views which are represented in more than one aspect, they can be exploited to establish relations between the aspects.

7.1.1 Approximating Aspect Distances

Let $a$ and $b$ be two $K$-dimensional aspects with representational graphs $g^a$ and $g^b$, mean correspondence mappings $\bar{M}^a$ and $\bar{M}^b$, and eigenvector-matrices $P^a$ and $P^b$ respectively. Then a view which is contained in both $a$ and $b$ can be represented by a point in a $K$-dimensional Euclidean space for each of the two aspect representations (see section 6.2.1). The point coordinates are derived from the correspondence mappings $X^a$ and $X^b$ from the representational graphs $g^a$ and $g^b$ associated with the view. They yield the following
coordinates in the affine subspaces of both aspects.

\[
\tilde{\xi}^a = \left( (\tilde{X}^a - \tilde{M}^a)^T \tilde{P}_1^a, \ldots, (\tilde{X}^a - \tilde{M}^a)^T \tilde{P}_K^a \right)^T \tag{7.1}
\]

\[
\tilde{\xi}^b = \left( (\tilde{X}^b - \tilde{M}^b)^T \tilde{P}_1^b, \ldots, (\tilde{X}^b - \tilde{M}^b)^T \tilde{P}_K^b \right)^T \tag{7.2}
\]

It must be noted that the dimensionalities of \(\tilde{\xi}^a\) and \(\tilde{\xi}^b\) are equal, namely \(K\), because \(K\) is the number of the object’s degrees of freedom and as such must be constant on the whole view manifold (although in cases where an object cannot be described in terms of one manifold, \(K\) might vary between the manifolds, see section 5.1). On the other hand \(\tilde{X}^a\) and \(\tilde{X}^b\) may have completely different dimensionalities, because they are determined by the number of nodes of \(g^a\) and \(g^b\) and are therefore of a fairly arbitrary nature.

If a whole set of common views \(O_{ab}\) is given one can estimate the centers of gravity \(\tilde{\sigma}^a, \tilde{\sigma}^b\) in the two aspect coordinate systems.

\[
\tilde{\sigma}^a = \frac{1}{N} \sum_{n \in O_{ab}} \tilde{\xi}^a_n \tag{7.3}
\]

\[
\tilde{\sigma}^b = \frac{1}{N} \sum_{n \in O_{ab}} \tilde{\xi}^b_n \tag{7.4}
\]

where \(N\) is the number of views in the overlap. One can now give a crude approximation to distance \(\Delta_{ab}\) between the origins of the two aspects \(a\) and \(b\) by

\[
\Delta_{ab} = \|\tilde{\sigma}^a\| + \|\tilde{\sigma}^b\|. \tag{7.5}
\]

Given the distances between all overlapping aspects, the distance between non-overlapping aspects can be approximated by the shortest connecting path of overlapping aspects. For example if aspects \(a\) and \(c\) are not overlapping, but the distances \(\Delta_{ab}\) and \(\Delta_{bc}\) to the aspect \(b\) can be estimated, \(\Delta_{ac}\) is estimated by \(\Delta_{ac} = \Delta_{ab} + \Delta_{bc}\).

If a distance can be directly estimated by the mutual overlap of two aspects and an indirect connecting path, the distance is taken to be the shorter of the two distances. Thus all violations of the triangle inequality are eliminated. This way a metric is imposed upon the aspects.

In case the set of aspects is composed of two or more completely unconnected subsets, these are treated separately in all following considerations and are said to constitute different object hypotheses.

It is important to note that this somewhat awkward distance measure has unique qualities rendering it fundamentally different from all other previously proposed methods. The distances are measured solely inside the view manifold, no reference is made to a space embedding the view manifold. This distinguishes it from other methods which retrieve perceptual manifolds in an unsupervised fashion, such as classical neural network
approaches or more recent techniques, e.g. (Tenenbaum, 1998; Tenenbaum et al., 2000). Even most techniques which establish the topology of views in a supervised fashion such as (Murase and Nayar, 1995; Beymer and Poggio, 1996; Tenenbaum and Freeman, 2000; Okada, 2001) require an embedding space in which distances can be measured. This point is so important, because a space embedding all views of an object requires finding a view representation suitable for the whole viewsphere of an object beforehand. This in turn raises difficult questions regarding the missing data problem (Okada, 2001), caused for example by self-occlusion, or it requires normalization procedures in order to render the view representation comparable even across large changes in viewpoint (Murase and Nayar, 1995; Beymer and Poggio, 1996).

7.1.2 Global Parameterization

So far a local topology was established between all views within an aspect and the distances between the single aspects were approximated. Still missing is a global parameterization of all views. Although a local parameterization of the views in an $K$-dimensional Euclidean space via the aspects representation was already obtained, it is not clear whether the global parameterization can also be embedded in a $K$-dimensional Euclidean space and still reflect the topology of views correctly. For example, although a ring can be parameterized locally in terms of a 1-dimensional Euclidean space, its overall topology can be accurately reflected only in a 2-dimensional Euclidean space.

The global parameterization space must therefore be estimated separately. To this end metric multi-dimensional scaling (see appendix B) is applied to the approximate distance matrix $\Delta$ of the aspects. Given a distance matrix between a number of objects, in our case the aspects, metric multi-dimensional scaling yields a representation of these objects as points in an $L$-dimensional Euclidean space, such that the distances between those points approximate the original distance matrix. Very similar to principal component analysis each dimension in this Euclidean space is associated with an eigenvalue, such that the appropriate dimensionality $L$ can be estimated by the previously introduced Scree-test (see section 6.2.4).

This way a point $\vec{c}^a$ in an $L$-dimensional Euclidean space $\mathcal{G}$ is assigned to each aspect $a$. But as this only defines coordinates in the global parameterization space $\mathcal{G}$ for the origin of the local aspect parameterization, one has to take one more step, namely to estimate a mapping for each aspect from the local parameterization to the global one.

7.1.3 Aligning the Aspects

In order to retrieve mappings from the local parameterization of each aspect into the global parameter space the already established global coordinates $\vec{c}^a$ of the aspect origins are used to determine the rough position of views and the overlap sets $O_{ab}$ are again exploited to yield the correct orientation of the different aspect mappings.
Figure 7.1: Neighboring Aspects: Shown is a schematic drawing of two neighboring aspects $a$ and $b$. The coordinates of the origins of the aspect representation are given by $\tilde{c}^a$ and $\tilde{c}^b$, respectively. The set of views in the overlap of $a$ and $b$ is denoted $O_{ab}$. The center of gravity of $O_{ab}$ in the local coordinate systems of $a$ and $b$ is given by $\tilde{o}^a$ and $\tilde{o}^b$.

Let $\tilde{c}^a$ and $\tilde{c}^b$ be the global coordinates of two overlapping aspects $a$ and $b$, and let $\tilde{o}^a$ and $\tilde{o}^b$ be the center of gravity of the overlap set $O_{ab}$ in local coordinates of $a$ and $b$ respectively. In this situation the global coordinates $\tilde{\mu}^{ab}$ of the center of gravity of $O_{ab}$ is estimated by

$$\tilde{\mu}^{ab} = \tilde{c}^a + \frac{\|\tilde{o}^a\|}{\|\tilde{o}^a\| + \|\tilde{o}^b\|} \left(\tilde{c}^b - \tilde{c}^a\right).$$

In doing so it was assumed that $\tilde{o}^a$ and $\tilde{o}^b$ are mapped onto the same point $\tilde{o}^{ab}$ and moreover that the mapped origins $\tilde{c}^a$, $\tilde{c}^b$, and $\tilde{\mu}^{ab}$ lie on a straight line (see figure 7.1). Additionally, it was assumed that the distance between two views is the same no matter in which aspect metric they are measured.

Assuming now that the mapping between the aspect $a$ and the global parameter space can be approximated linearly, a matrix $A_a$ can be estimated from

$$\tilde{\mu}^{ab} - \tilde{c}^a = A_a \tilde{o}^a.$$

$A_a$ is a $L \times K$ matrix, where $L$ is the dimensionality of the global parameterization space $G$ and $K$ is the dimensionality of the local parameterization, or in other words the dimensionality of the view manifold. One such equation can be obtained for each overlap of $a$ and another aspect. Consequently, $K$ overlap sets are sufficient to determine $A_a$. If more are available a solution optimal in a least-square sense can be found by standard techniques. If fewer are available, the aspect representation cannot be incorporated into the global parameterization and must be discarded. This, however, is rarely the case as the required number of overlapping aspects is just the number of the physical degrees of freedom of the object.
So, if a new view, represented by an image, is to acquire coordinates \( \tilde{\xi} \) in terms of the global parameterization, all representational graphs \( g^a \) are matched on the image. The one with the highest similarity is then used to assign to the new view local coordinates \( \tilde{\xi}^{a_{\text{max}}} \) in terms of the aspect representation. These are then transformed into the global coordinate system by
\[
\tilde{\xi} = A_{a_{\text{max}}} \tilde{\xi}^{a_{\text{max}}} + \tilde{\epsilon}^{a_{\text{max}}}.
\] (7.8)

While deriving the expressions above a number of relatively crude assumptions were made. Most of those are fairly harmless as they are rooted in the basic assumption that the aspects are small compared to the curvature of the underlying manifold, which they are by virtue of construction.

However, requiring that the distances measured in two different aspect representations have to be the same in this context a very strong assumption, especially as we have so fiercely argued in section 7.1.1 that the independence of aspects is an important feature of our approach. But as the whole process is set up to be fairly robust, it is possible to somewhat relax this requirement. It probably suffices if the metric does not change more drastically between two neighboring aspects than the views themselves such that any change in representation is induced by a change in the visual appearance of the view.

### 7.1.4 Summary

All the previously described processing stages can now be integrated into one recursive algorithm which derives a complete object representation from single views.

If views of an object are presented to the system one by one in no particular order, the algorithm proceeds the following way. First a view is associated with an aspects, describing the transformation properties of the object by a local linear model. These linear charts to the actual view manifold are then integrated into one global embedding parameter space by determining their topology from the mutual overlap.

So, for each view of an object which is presented to the system, the following steps are taken:

1. **Initialization:** If this is the first view and no aspect representation has so far been created, a graph is extracted from the view, after prior segmentation (see section 6.1.1). This graph is then the representational graph \( g^a \) of a new aspect \( a \).

2. **Aspects:** All representational graphs \( g^a \) associated with local aspect representations \( a \) are matched onto the view, as described in section 6.1.2.

   (a) If the similarity between the new view and a representational graph \( g^a \) exceeds the threshold value \( t \), i.e. \( s_{\text{min}} > t \), the view is assumed to be well represented in the associated aspect \( a \). The retrieved correspondence mapping \( \tilde{X}^a \) between graph and view is stored and the linear aspect model is updated by recalculating the principal components of the correspondence mappings (see section 6.2.1).
(b) In case a view is represented well by two aspects $a$ and $b$ the view belongs to the overlap set $O_{ab}$ of those two aspects, and they are said to be neighboring.

(c) If the view cannot be represented in any of the current aspects, a graph is extracted from the view, after prior segmentation (see section 6.1.1). This graph is then the representational graph $g^a$ of a new aspect $a$.

3. **Object Hypotheses:** Aspects that are neighboring or can be connected by a succession of neighboring aspects are grouped together to form an object hypothesis. An object hypothesis is thus a coherent patch, or atlas, of an object’s view manifold for which neighborhood relations between the linear models as well as distances between the aspects in the sense of section 7.1.1 are defined.

4. **Global Parameter Spaces:** A global parameter space $G$ is now assigned to each object hypothesis. The mapping between the origins of the linear aspect representations and the new global coordinate system is retrieved via multi-dimensional scaling (see section 7.1.2).

5. **Mappings from Global to Local Representations:** The transformations $A_a$ from the local linear parameterizations of all aspects $a$ into the embedding global parameter space $G$ are retrieved by once again exploiting the overlap between the aspects (see section 7.1.3).

In this fashion the object representation can be refined as more and more views become available. The process has no well defined point of termination, because it cannot be guaranteed, that all degrees of freedom of one object are covered for any given representation. But when only views are presented to the system, which are already well represented, the object representation should slowly converge towards a stable solution.

### 7.1.5 Experiments

The algorithm was tested separately on image databases of six different objects, two plastic figures, an ink blotter, an animal, a pig, and a box (see figure 7.2). The views were presented to the system by selecting a view at random in each iteration step. Multiple presentations of the same view were not prohibited.

**Database**

Four of the six objects were taken from already existing object databases. The two plastic figures of Tom and a dwarf were recorded by Gabriele Peters (Peters, 1999). Views were recorded on a digitally controlled turntable by varying the longitudinal angle in steps of $3.6^\circ$ from $0^\circ$ to $360^\circ$ and the latitudinal angle in steps of $3.6^\circ$ from $0^\circ$ to $90^\circ$, thus yielding a total of 2500 views reflecting two degrees of freedom.
7.1. FROM CHARTS TO MANIFOLDS

Two objects, the animal and the ink blotter, were created by the ray tracer POVRAY. The generated views were derived from the same part of the viewsphere (the upper hemisphere) as for the two plastic objects, sampling distance was 3°. The number of views generated per object is 3600.

The last two objects used, the pig and the box, are part of the Columbia database, made available by the computer vision group of the University of Columbia (Murase and Nayar, 1995). Here, the objects are rotated only around one axis in steps of 5°, resulting in 72 views per object.

Results

Figures 7.4 – 7.9 show the resulting object representations for the six objects. The results are displayed by showing one view per aspect in the embedding parameter space \( \mathcal{G} \) by projecting the manifold in two orientations onto the image plane. Overlapping aspects are connected by a line.

Figure 7.3 shows the evolution of the number of aspects and object hypothesizes as a function of views presented for those sets of views reflecting two degrees of freedom. After 1000 – 1500 presentations the final number of aspects is nearly reached, all remaining views are already represented by the existing aspects. After another 500 presentations most aspects are integrated into one coherent object representation.

For three of the six objects (see figures 7.5, 7.7, 7.4(a)) the derived topology and embedding into a three-dimensional space the nature of the underlying transformation is well captured. Views which are located in close proximity on the view sphere are associated with similar coordinates in the embedding space and vice versa.

The view topology, derived for the dwarf object (figure 7.9), is less intuitive. Here the view topology, as for example given by the relative viewing angle, is not well established. The reason is to be found in the asymmetric properties of the similarity function used. To determine whether a certain view is part of a given aspect, the representational graph is matched onto the view and all subsequent evaluations are solely based on the resulting similarity. But to yield a high similarity it suffices if the representational graph is similar
Figure 7.3: **Evolution of Representations:** For four example objects with training images containing two degrees of freedom the change in the number of aspects and the number of estimated objects are shown as more and more images are presented. After 1000 to 1500 presentations most aspects seem to have a corresponding representation. At about the same time the representation of single aspects are fused to form a small set of integrated object representations. After about 2000 presentations all but one or two aspects are fused to from a complete object representation.

to a *part* of the view. In the case of the dwarf, the top view, which shows only the hat, does match the hat of the side view of the dwarf very well and the views are thus assumed to be of the same object aspect. This could be trivially avoided by employing a symmetric similarity function (for an example see (Wieghardt and von der Malsburg, 2000)), which requires the similarity between the top of the dwarf to the side of the dwarf to be the same as vice versa. Such a similarity function was not used as it would require the solution of the correspondence problem in two directions and thus double the necessary calculation time, which was already fairly long. But it must be emphasized that the poor result for the dwarf is due to a detail in the implementation rather than due to the general approach.

More general problems are exhibited in figures 7.8 and 7.4(b). In both cases the topology of views is not well represented. Views which, although similar, are not neighbors in physical terms, are represented as being part of the same aspect of the object. This in turn yields a topology which, as discussed in section 5.1, cannot be realized as a manifold. This is so because the assumed virtual transition between the physically unconnected views locally introduces a new degree of freedom, which is not present at other points of the view sphere. In the examples presented this effect might be avoided by employing a more sensitive similarity measure or by resorting to images of higher resolution. But this
problem cannot be avoided in general. It is solely caused by using the similarity of single
to determine their topology.

Lastly we acknowledge that the retrieved representations of the upper half of the
does not look at all like a sphere. The more cylindrical shape is due to our
decision not to compensate for rotation in plane explicitly (see 6.2.1).
Figure 7.4: Full View Topology (Pig and Box)
Figure 7.5: **Full View Topology (Tom)**: Displayed are views from each aspect in the object representation in the xy-plane of the embedding parameter space. Overlapping aspects are connected by a line.
Figure 7.6: **Full View Topology (Tom)**: Displayed are views from each aspect in the object representation in the xz-plane of the embedding parameter space. Overlapping aspects are connected by a line.
Figure 7.7: Full View Topology (Ink Blotter)
Figure 7.8: **Full View Topology (Animal)**: The system confuses back and front of the animal and therefore a virtual transition is created.
Figure 7.9: **Full View Topology (Dwarf)**: The underlying view topology is only partly captured by the system. Confusion between the top and the side view of the dwarf’s hat distorted the topology.
CHAPTER 7. VIEW MANIFOLDS

7.2 Learning to Estimate Pose Angles

The main reason to organize the views in some topological order according to similarity was to find a parameterization of the underlying view manifold. The claim was that the topology of views provided by the parameterization of the view manifold facilitates the learning of mappings between the visual appearance of an object and properties related to physical state. In order to illustrate this, the mapping between views and viewing angles, which are certainly an important property of the physical state, will be learned from very few examples. As discussed in chapter 5 such a parameterization of views in terms of physical object parameters is very desirable, as it facilitates interaction with the object such as, e.g., in tasks requiring visually guided grasping.

We therefore try to map the viewing angles onto the derived view representations with a small number of support views. A support view is a view for which the viewing angles are given, in our case from ground truth. In a more general framework they might be obtained from interactive manipulation of the object.

In order to associate the viewing angles with the visual appearance of the object a linear interpolation scheme is set up on the view manifold.

7.2.1 Approximating Distances

The first question which arises in the context of linear interpolation is, “How are distances between views on the manifold estimated?”. Although a metric is given for views within a single aspect, although the relations between different aspects are part of the object representation, and although a global topology was established on the views, the problem of measuring distances on the view manifold was so far not addressed.

We here propose a distance measure between two views $n$ and $m$ with global coordinates $\tilde{c}_n$ and $\tilde{c}_m$, respectively, which on the one hand provides a good approximation to the actual distance and on the other hand is computationally not too expensive and converges to a solution in all cases.

To obtain the distance, the connecting line between $\tilde{c}_n$ and $\tilde{c}_m$ through the embedding parameter space is projected piecewise onto the linear aspect representations in the embedding space. The sum of the projected pieces is then the estimated distance.

A point $\tilde{\zeta}$ in the global parameter space $\mathcal{G}$ is projected onto the closest aspect representation $a_0$

$$a_0 = \arg\min_a \left\{ \| \tilde{\zeta} - \tilde{c}_a \| \right\}$$  \hspace{1cm} (7.9)

where $\tilde{c}_a$ is the global position of the origin of the aspect representation $a$ and $\| \cdot \|$ is the Euclidean distance in $\mathcal{G}$, as follows:

$$\mathcal{P} \left( \tilde{\zeta} \right) = A_{a_0} A_{a_0}^* \left( \tilde{\zeta} - \tilde{c}_{a_0} \right) + \tilde{c}_{a_0}$$  \hspace{1cm} (7.10)

$$A_{a_0}^* = \left( A_{a_0}^T A_{a_0} \right)^{-1} A_{a_0}^T$$  \hspace{1cm} (7.11)
7.2. LEARNING TO ESTIMATE POSE ANGLES

$A^\star_{ao}$ is the Moore-Penrose inverse of $A_{ao}$. The distance between $\tilde{\zeta}_n$ and $\tilde{\zeta}_m$ can now be approximated by the sum of piecewise projected line segments of the connecting line through the embedding space.

$$d (\tilde{\zeta}_n, \tilde{\zeta}_m) = \sum_{i=1}^{M} \left\| P \left( \left( \frac{\tilde{\zeta}_n - \tilde{\zeta}_m}{M} + \frac{\tilde{\zeta}_m}{M} \right) \right) - P \left( \left( \frac{\tilde{\zeta}_n - \tilde{\zeta}_m}{M} + \frac{\tilde{\zeta}_m}{M} \right) \right) \right\|$$  \hspace{1cm} (7.12)

where $M$ is number of pieces used for the projection. In all experiments $M = 1000$ yielded satisfactory results.

7.2.2 Interpolation on the Manifold

Assuming now that for a set of views $V$ in addition to their coordinates in $G$ the physical viewing angles are supplied, from a source outside vision, how can the viewing angles be estimated for all views in the object representation?

Because structure can be imposed upon the views, as shown before, and on the viewing angles, simple linear interpolation can be used to infer the viewing angles for all views. So, given a view with coordinates $\tilde{\zeta}$ and the $K+1$ nearest neighbors of $\tilde{\zeta}$ in $V$: $\tilde{\zeta}_1, \ldots, \tilde{\zeta}_{K+1}$, where $K$ is the dimensionality of the view manifold, and given their associated viewing angles $\tilde{\alpha}_1, \ldots, \tilde{\alpha}_{K+1}$, a linear mapping $B$ can be obtained, from

$$\tilde{\alpha}_2 - \tilde{\alpha}_1 = B d (\tilde{\zeta}_2, \tilde{\zeta}_1) \frac{A_1 A_1^* (\tilde{\zeta}_2 - \tilde{\zeta}_1)}{\left\| A_1 A_1^* (\tilde{\zeta}_2 - \tilde{\zeta}_1) \right\|}$$ \hspace{1cm} (7.13)

$$\vdots$$

$$\tilde{\alpha}_{K+1} - \tilde{\alpha}_1 = B d (\tilde{\zeta}_{K+1}, \tilde{\zeta}_1) \frac{A_1 A_1^* (\tilde{\zeta}_{K+1} - \tilde{\zeta}_1)}{\left\| A_1 A_1^* (\tilde{\zeta}_{K+1} - \tilde{\zeta}_1) \right\|}$$ \hspace{1cm} (7.14)

where $A_1$ is the linear mapping between the local representation of the aspect which best represents the view belonging to $\tilde{\zeta}_1$ and the global parameterization space. This way unknown view angle $\tilde{\alpha}$ of $\zeta$ is given by

$$\tilde{\alpha} = B d (\tilde{\zeta}, \tilde{\zeta}_1) \frac{A_1 A_1^* (\tilde{\zeta}_1 - \tilde{\zeta}_1)}{\left\| A_1 A_1^* (\tilde{\zeta}_1 - \tilde{\zeta}_1) \right\|} + \tilde{\alpha}_1$$ \hspace{1cm} (7.15)

This is basically a very straightforward linear interpolation scheme, with the exception that all values must be measured within a potentially curved manifold. The directions, e.g. $\tilde{\zeta}_1 - \tilde{\zeta}_1$, were therefore measured in the tangent space to the approximation to the view manifold, namely the local linear chart of the aspect to which $\tilde{\zeta}_1$ belongs, and the lengths had to measured within the manifold by the distance measure provided by equation 7.12.

In the above calculation it was assumed that $B$ is well defined and thus that neither the $\tilde{\zeta}_n$ or the $\tilde{\alpha}_n$ are linearly dependent. In general this can cannot be guaranteed, but as this
case turned up very infrequently in the experiments below a very simple measure was taken to evade such situations. In case the $\tilde{\zeta}_n$ or the $\tilde{\alpha}_n$ do not define a $K$-dimensional affine subspace, the $K+1$th neighbor is discarded and replaced by the $K+2$th nearest neighbor. We acknowledge that this is not a clean solution to the problem, but as this section is only to determine whether constructing a view manifold before learning facilitates the process of association, and the point can be made with this primitive learning algorithm, we believe it sufficient. More sophisticated learning algorithms taking advantage of prior established topologies can be found in (Walter, 1996). Although the problem of irregular sampling as present in this case is not discussed in detail there, it gives an impression how to properly take topologies into account in the paradigm of learning and interpolation.

### 7.2.3 Experiments

The interpolation scheme was tested for the four objects introduced in section 7.1.5. For each object representation 40 views were picked at random and were labeled with their ground truth angle values. From the sets used to generate the representations an equally spaced grid in angle space of views was selected as test views. The grids contained roughly a quarter of the original sets.

Each test view had to be classified in terms of its relative viewing angle by first assigning global coordinates $\tilde{\zeta}$ to it by matching it against all representational graphs and successive interpolation using a subset of the randomly chosen 40 ground truth views. The resulting mean and median classification errors are shown in figure 7.10 as a function of the number of available ground truth data. Given that the topology of views is well reconstructed, very few ground truth samples suffice to create a reliable mapping between views and angles. For the Tom and ink blotter object as few as 20 samples are necessary to achieve an accuracy of around $15^\circ$, and an accuracy of up to $7^\circ$ mean error was possible for 40 samples. This is remarkable when taking into account that only image similarities were used to set up the view manifold. An accuracy of $5^\circ - 20^\circ$ in pose estimation should be sufficient for most applications, such as visually guided grasping. In case of the other two objects substantial mean classification errors still remain even for 40 examples, although the median error is nearly as low as for the ink blotter and Tom objects. This effect is caused by local distortions of the view topology. Views perceived to be neighbors are not neighboring in angle space. This yields fatal interpolation errors, which are confined to regions on the view sphere where the local topologies differ. For the animal object, where a single virtual transition is introduced, the error can be dramatically reduced by using more and more samples, because the region in which interpolation across the virtual transition is required is gradually reduced by adding new samples. Outside these error regions the performance is, also for those objects, acceptable, as shown in figure 7.11. All in all the mean classification errors are comparable with those of algorithms based on view topologies given \textit{a priori} (Peters, 2001).
7.2. LEARNING TO ESTIMATE POSE ANGLES

![Graph showing mean and median error as a function of the number of support views for different objects.](image)

(a) Mean Error

(b) Median Error

Figure 7.10: **Interpolation Error**: Shown is the mean (a) and median (b) error of interpolation as a function of the number of ground truth points for different objects.
Figure 7.11: The Distribution of Errors: Shown is the classification error over the ground truth angles ranging from $0^\circ$ to $360^\circ$ and from $0^\circ$ to $90^\circ$. The plotted triangles show the locations of the available ground truth data from which the other pose angles were to be inferred.
Chapter 8
Discussion and Outlook

It was demonstrated that it is feasible to derive an appearance-based object representation from nothing but single pictures. The approach taken went all the way from the single views of an object to aspects to a completely integrated object description. Following this path of successively higher integration makes it possible to set up an online learning scheme for pose-angles, which demonstrates the systems ability to go beyond the visual appearance and capture fundamental properties of an object’s three-dimensional nature.

At each stage of integration the entities of representation (model graphs, aspects, view manifolds) provide new knowledge about an object. The description of single views in terms of representational graphs enables the system to recognize known views in the presence of slight variations by elastic graph matching without any further knowledge about the overall structure. The representation of aspects in terms of representational graphs and linear transformation properties helps to constrain the correspondence problem in a number of scenarios as demonstrated in chapter 3 and provides a local topology of views, which in turn is a basis on which to locally parameterize the views. The global representation, finally, integrates the local topologies and parameterizations into one global topology-preserving parameterization, which provides a basis for the rapid learning of pose-angles. Moreover, it was shown that creating the global parameterization based upon the the already established aspect parameterizations helps to confine the destructive effects of self-similarity of an object to small parts of the viewsphere, still yielding valid results even in those cases where the views of an object cannot be described in terms of a single manifold in appearance space.

Starting from a simple view representation and establishing local topologies by aspect representations before integration in terms of a global parameterization also eliminated the need for an embedding space for all views. In the exemplary case of a representation by graphs creating an embedding space would imply that all views must be dealt with in terms of a single graph. Such a graph would unavoidably face the problem that some of its nodes disappear from the field of view while the object is rotating. The task of handling those invisible parts which are nonetheless still part of the representation was
termed the *missing data problem* in (Okada, 2001). By utilizing completely independent representations for each aspect of the object we here avoided the *missing data problem*. We also kept the representational space low-dimensional at all times, because invisible parts of the object were simply not represented.

The independence of the individual aspects has, of course, its limits, because we still need to relate distances measured in neighboring aspects to each other. Therefore the representation of neighboring aspects should not change more drastically than required by the disappearance and reappearance of the graph nodes.

### 8.1 Biological Point of View

Relating computational models to neurophysiological and psychophysical data is notoriously difficult. The similarities between the data retrieved from biological systems and a computational model are easily overinterpreted and are often misleading. Keeping this qualifying remark in mind, we will nonetheless point out some parallels between the model presented here and biologically relevant findings.

One of the central features of our construction was the topological ordering of object appearances. The evidence for the general notion that object representations in the mammalian visual system are largely view- or appearance-based, is overwhelming (Bülthoff and Edelman, 1992; Edelman, 1997; Wallis and Bülthoff, 1999). Moreover it was shown in (Wang et al., 1996) that representations in the inferotemporal region of the macaque monkey are not only view based but also exhibit a topological ordering in terms of the viewing angle.

Interesting parallels can also be found between our proposed model of object representation and the finding reported in (Farrell and Shepard, 1981; Shepard and Farrell, 1985). Here human performance was evaluated in terms of reaction times and estimated transformation times related to two distinct tasks. The first task required the discrimination between two views of the same object, while the second required implicit transformation between the two views in an apparent motion task. By measuring reaction and transformation times the following observation was made:

> [...] whereas transformation times increased (often linearly) with distance, discrimination times decrease non-linearly and asymptotically (towards a constant) with distance. Second [...] the relevant distances are different in the two cases, being geodesic distances within the curved manifold of orientations in the first case and distances through the embedding space in the second. (Shepard and Farrell, 1985)

This distinction between discrimination and transformation is well reflected in our model. Discrimination implemented by elastic graph matching shows exactly the reported qualities, if the similarities as measured by elastic graph matching are taken to be a measure
of task-complexity comparable to the one obtained from reaction times. The similarities are bounded between 0 and 1, and thus show an asymptotic behavior in the presence of decreasing similarity, and the similarity of elastic graph matching is strictly view-based and takes no notice of the view manifold. This might be interpreted as measuring distances by cutting through some embedding space. On the other hand in tasks, which require transformations of views, such as interpolation, distances are measured within the view manifold, just as reported in the psychophysical experiments above. This way our view-based approach to object representation and the distinctions made between the similarities of single views and the overall structure is well founded.

8.2 Continuity in Time

Regarding the actual process of bootstrapping the representation from an assortment of single views, one might argue, that we have complicated an already complex problem, as different views of an object usually occur in a given context, which supplies additional information, but is completely ignored by our approach.

An issue often raised in this context is that of continuity in time, i.e., that views do not occur in isolation but in temporal sequences, which in turn can be exploited to establish a view topology at almost no cost. It has indeed been shown in a series of psychophysical experiments that continuous sequences of views are much better suited to yield a coherent object representation than single isolated ones (Kellman, 1984; Kellman and Short, 1987; Wallis and Bülthoff, 1999). So in how far could such additional information be used in the framework developed above?

So far continuity in time has only been used in artificial models to solve the association problem, i.e. the question of whether or not two views correspond to the same object (Rolls, 1994; Nestor and Rolls, 1998; Becker, 1999). From the results obtained in chapter 7 two additional points can be identified, where the continuity in time might play a vital role in establishing an object representation – establishing a coherent metric and eliminating virtual transitions.

As the single aspects are initially treated independently and the metric within each aspect is solely based upon variation of visual appearance, the metric established upon the single views might vary strongly in its relation to the metric of the underlying physical transformation parameters. Observing a smooth motion in time, which transits through a number of aspects, might help to realize those differences and might also allow to compensate for them by, for example, requiring constant perceived speed in all aspects.

As pointed out before the strictly view-based approach suffers from the fundamental inability to separate very similar views which are not neighbors in term of the physical parameterization. This in turn causes problems when relating the retrieved appearance-based representation to the physical world, as it was shown in section 7.2. To avoid these points of virtual transitions, where views seem to be neighboring which in reality are
not, the observation of continuity in time might be exploited. But as the observation of continuous motion only provides a one-dimensional topology on the views, it is still required to associate similar views with each other. In other words, continuous sequences of views tell the system which views are neighboring, but it does not say which views are not neighboring. So the system still requires the similarity-based approach to create the full topology.

If, on the other hand, a point of virtual transition could be identified as such and a decision is required which of two possible transitions is real, a continues transformation along one of these transitions would allow to validate the real transition and discard the other. But this case only applies if additional information about the topology of the view manifold is available. Given this information a point of virtual transition can be identified because it locally violates the topological structure. In the case of rigid objects, as opposed to objects with internal degrees of freedom, the topological structure in appearance space is always the same, because the topology is then not a property of the object but of the three-dimensional space itself. For rigid objects the topology is always that of sphere or torus, which depends on the fact whether in-plane rotated views are identified which each other or not. In such a case violations are easily detected.

So from our results we would argue that continuous sequences of views can help to establish an object representation, if the motion is smooth and is accompanied by additional knowledge about the general topology of views, which for rigid objects is always the same.

## 8.3 Active Handling of Objects

Even more accurate information about the state of the world can, of course, be obtained if it can be physically manipulated, e.g., if an object can be picked up and rotated, such that visual appearance can at all times be related to a physical orientation (see, e.g., (Kefalea et al., 1999)). On the other hand one can not assume that all our information about the relations between visual and physical states of the world are acquired this way as it is a very time consuming and involved process, and as many objects cannot be handled or walked around.

The experiments discussed here suggest two possible areas where information retrieved from actual physical interaction might be crucial – resolving local ambiguities and acquiring general knowledge about the topology of appearance manifolds. Resolving the ambiguities is in so far ideally suited to be dealt with by active manipulation of an object as it requires only very little and well determined movements.

Whereas the ambiguities can also be resolved by other means, as described above, the general topology of transformation manifolds can only be retrieved if a direct relation between the visual information and the physical world exists. Only by direct interaction it is possible to realize that all objects independent of the visual appearance are well
described on a closed manifold, which only upon rotation of multiples of 360° returns an object to its original orientation.

Active manipulation of the environment must therefore constitute a vital part of creating an internal model of the world.
Appendix A

Principal Component Analysis

Given a set of $P$ data vectors $\mathbf{x}_p \in \mathbb{R}^N$ the goal of principal component analysis is to find an affine subspace of dimensionality $K$ in $\mathbb{R}^N$ so that orthogonal projection of the data points onto this subspace changes them minimally in a least square sense. So an orthonormal basis $\mathbf{u}_n$ of $\mathbb{R}^K$ and a vector $\mathbf{m}$ is needed with

$$\tilde{\mathbf{x}}_p = \sum_{n=1}^{K} z_{pn} \mathbf{u}_n + \mathbf{m}$$  \hspace{1cm} (A.1)

and

$$E(\mathbf{u}_1, \ldots, \mathbf{u}_K, \mathbf{m}) = \frac{1}{P} \sum_{p=1}^{P} (\tilde{\mathbf{x}}_p - \bar{\mathbf{x}})^2 = \text{min.}$$  \hspace{1cm} (A.2)

With $\bar{\mathbf{x}}$ being the mean of the $\tilde{\mathbf{x}}_p$

$$\bar{\mathbf{x}} = \frac{1}{P} \sum_{p} \tilde{\mathbf{x}}_p$$  \hspace{1cm} (A.3)

and $\mathbf{C}$ the covariance matrix

$$\mathbf{C} = \frac{1}{P-1} \sum_{p} (\tilde{\mathbf{x}}_p - \bar{\mathbf{x}}) (\tilde{\mathbf{x}}_p - \bar{\mathbf{x}})^T$$  \hspace{1cm} (A.4)

the $\mathbf{u}_n$ are given by the $K$ normalized eigenvectors of $\mathbf{C}$ with the largest eigenvalues $\lambda_n$

$$\mathbf{C} \mathbf{u}_n = \lambda_n \mathbf{u}_n$$  \hspace{1cm} (A.5)

and the $\mathbf{m}$ is given by

$$\mathbf{m} = \bar{\mathbf{x}}$$  \hspace{1cm} (A.6)

Furthermore a relation between the eigenvalues $\lambda_n$ and the error criterion can be derived

$$E = \sum_{n=K+1}^{N} \lambda_n$$  \hspace{1cm} (A.7)
Figure A.1: Geometric Interpretation of Principal Component Analysis: If the approximate probability density of the input data takes the shape of ellipsoid, then principal component analysis can be expected to retrieve the center and axes of this ellipsoid. The the ratio of the square roots of the eigenvalues given estimate of ratio of the axes of the ellipsoid. An example for a two dimension configuration is shown.

For a detailed proof see for example (Bishop, 1995). Thus the best approximation by a affine subspace is achieved, if the $u_n$ are ordered so that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_N$. This way a hierarchy of affine subspaces is given by

$$\tilde{x}^K = \sum_{n=1}^{K} \tilde{z}_n \tilde{u}_n + \tilde{x}$$

(A.8)

that allow to represent the data for any given dimension $K$ optimally (as defined by equation A.2).

After having obtained this hierarchy of affine subspaces it is often desirable to determine the most suitable value for $K$. In this context it is often assumed that the distribution of data points is influenced by two separate processes, the first being related to degrees of freedom within the data and the second stemming from noise caused, e.g., by measurement-errors. In such cases one is interested in separating the variations stemming from internal degrees of freedom from those caused by noise.

In this thesis a modified version of the Scree-test (Fahrmeir and Hamerle, 1984) is employed to this end, which makes two basic assumptions. Firstly the variations due to internal degrees of freedom are large compared to those stemming from noise. And secondly the noise is modeled well by a Gaussian process. According to (Cattel, 1966) the eigenvalues of an empirical covariance matrix caused by independently identically
Figure A.2: Scree-Test: Figure A.2(a) shows the eigenvalues for a set of 40 data point in a 10 dimensional space, drawn from a distribution Gaussian distribution with mean deviation of 2.5 along the first dimension, 2 along the second and 1 along all others. Figure A.2(b) shows the corresponding values of $\frac{\lambda_{i+1}}{\lambda_i}$ distributed noise follow a flat approximately exponential decline. We thus take

$$K_0 = \arg \min_i \frac{\lambda_{i+1}}{\lambda_i}$$

(A.9)

to be the suitable value for $K$ (see figure A.2)
Appendix B

Metric Multidimensional Scaling

The task in multi-dimensional scaling is to find a configuration of \( N \) points \( \vec{x}_i \in \mathcal{R}^k \), when given only a \( N \times N \) distance matrix \( \Delta_{ij} \), that contains the mutual distances between those points. The method discussed here is the classic version, metric multi-dimension scaling (Torgerson, 1952; Torgerson, 1958), as employed in chapter 7. For a discussion of different methods in a more general framework see (Fahrmeir and Hamerle, 1984; Mathar, 1997).

For \( \Delta \) to be a distance matrix the following condition must be met:

\[
\Delta_{ii} = 0 \quad (B.1)
\]

\[
\Delta_{ij} = \Delta_{ji} \quad (B.2)
\]

The classical metric multi-dimensional scaling is formulated as minimizing the expression

\[
\| E_N \left( \Delta^{(2)} - (\vec{x}_i - \vec{x}_j)(\vec{x}_i - \vec{x}_j)^T \right) E_N \|, \quad (B.3)
\]

where

\[
\Delta^{(2)} = (\Delta_{ij}^2) \quad (B.4)
\]

\[
E_N = I_N - \frac{1}{N} 1_{N \times N} = \begin{pmatrix}
1 - \frac{1}{N} & -\frac{1}{N} & \cdots & -\frac{1}{N} \\
-\frac{1}{N} & 1 - \frac{1}{N} & \cdots & -\frac{1}{N} \\
\vdots & \vdots & \ddots & \vdots \\
-\frac{1}{N} & -\frac{1}{N} & \cdots & 1 - \frac{1}{N}
\end{pmatrix} \quad (B.5)
\]

In case the matrix norm \( \| \cdot \| \) is invariant under orthogonal transformations, i.e. \( \| A \| = \| TAT^T \| \) for the orthogonal matrix \( T \), such as for example \( \| A \| = \sum_{ij} A_{ij}^2 \), the solution to the minimization problem can be found as follows.

If the matrix \( -\frac{1}{2} E_N \Delta^{(2)} E_N \) is diagonalized such that \( -\frac{1}{2} E_N \Delta^{(2)} E_N = T \text{diag}(\lambda_1, \ldots, \lambda_N) T^T \)

with an orthogonal matrix \( T = (\vec{t}_1, \ldots, \vec{t}_N) \) and corresponding eigenvalues \( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_N \), then a solution to

\[
\min_{\vec{x}_1, \ldots, \vec{x}_N \in \mathcal{R}^k} \| E_N \left( \Delta^{(2)} - (\vec{x}_i - \vec{x}_j)(\vec{x}_i - \vec{x}_j)^T \right) E_N \|
\]

(B.6)
is given by

\[ \bar{x}^T_1 = \left( (\lambda_1^+)^{\frac{1}{2}} t_{11}, \ldots, (\lambda_K^+)^{\frac{1}{2}} t_{k1} \right) \]

\[ \vdots \]

\[ \bar{x}^T_N = \left( (\lambda_1^+)^{\frac{1}{2}} t_{11}, \ldots, (\lambda_K^+)^{\frac{1}{2}} t_{kN} \right) \]  \hspace{1cm} (B.7)

The \( \lambda_k^+ \) are the positive eigenvalues of \(-\frac{1}{2}\bar{E}_N \Delta^{(2)} E_N\), the negative eigenvalues are “cut off” by setting them to zero. If no negative eigenvalues exist \( \Delta \) is Euclidean and the \( \bar{x}_n \) are also a solution to (Hartung and Elpelt, 1992):

\[ \min_{\bar{x}_1, \ldots, \bar{x}_N \in \mathbb{R}^K} \sum_{ij} \left\| \frac{\Delta^2_{ij}}{\bar{E}_{ii} \bar{E}_{jj}} - (\bar{x}_i - \bar{x}_j)(\bar{x}_i - \bar{x}_j)^T \right\| \]  \hspace{1cm} (B.8)

By increasing \( k \) and thus adding dimensions to the embedding space, the given distance matrix \( \Delta \) can be approximated with increasing accuracy. The relationship between the data points \( \bar{x}_n \) in embedding spaces of different dimensionality is given by a very simple orthogonal projection. To evaluate the choice of a concrete \( K \) for a given application, the distribution of eigenvalues \( \lambda_k \) can be used in the same fashion as in the case of principal components.
Appendix C

Zusammenfassung in deutscher Sprache

C.1 Einleitung


Zunächst muss die Möglichkeit bestehen, das Modell zuverlässig mit der Welt in Verbindung zu bringen, d.h. bekannte Objekte oder vielmehr bestimmte Ansichten bekannter Objekte müssen als solche erkannt werden können. Da man nicht annehmen kann, dass alle möglichen Ansichten eines Objektes gespeichert werden können oder auch nur bekannt sind, muss man verlangen, dass dieser Erkennungsprozess ein gewisses Maß an Verallgemeinerungsfähigkeit aufweist. Der Grad und die Art der Verallgemeinerung sind allerdings umstritten. Das im Folgenden verwendete System der elastischen Graphenanpassung beispielsweise verallgemeinert gut über kleine Objektdeformationen und Beleuchtungsänderungen. Andere Verfahren hingegen haben den Anspruch auch über große Variationen verallgemeinern zu können. Bei solchen Verfahren besteht allerdings das Problem, dass die Verallgemeinerungsfähigkeit zu Lasten der Diskriminierungsfähigkeit geht, d.h. dass auch Ansichten von verschiedenen Objekten als Variation der Ansicht eines einzelnen
Objektes wahrgenommen werden.

Des weiteren muss eine Objektrepräsentation Relationen zwischen den wahrgenommenen Ansichten erzeugen, die die Relationen reflektieren, welche durch die physikalische Welt gegeben sind. Bei dieser Eigenschaft spricht man von einem Isomorphismus zweiter Ordnung. Während ein Isomorphismus erster Ordnung im Zusammenhang mit Objektrepräsentationen eine Repräsentation durch Ähnlichkeit bedeutet, verlangt ein Isomorphismus zweiter Ordnung lediglich, dass die physische Ähnlichkeit von Objekten sich auch in der Ähnlichkeit ihrer Repräsentation widerspiegelt. Da zum Beispiel eine Katze mehr Merkmale mit einem Hund gemeinsam hat als mit einem Buch, kann man davon ausgehen, dass sich auch die Repräsentationen von Hund und Katze in irgendeinem Sinne ähnlicher sind als die von einem Buch und einer Katze. Übertragen auf die verschiedenen Ansichten eines Objektes entspricht die Forderung nach einem Isomorphismus zweiter Ordnung der Forderung nach einer Metrik auf den Ansichten, die die physikalische Nachbarschaft der Orientierungen widerspiegelt.

Die letzte wichtige Anforderung an jede Objektrepräsentation ist die der Lernbarkeit. Die Eigenschaften einer Objektrepräsentation, die von einem spezifischen Objekt abhängen, müssen aus der wahrgenommenen visuellen Information gewonnen werden können. Dies gilt sowohl für den Prozess der Erkennung von bekannten Ansichten als auch für die Erzeugung einer Metrik auf den bekannten Ansichten.

Diese Arbeit beschäftigt sich damit, wie Erkenntnisse über die Transformationseigenschaften von Ansichten gewonnen werden können, wie sie im Zusammenhang mit der Erkennung genutzt werden können und inwieweit sie eine Metrik auf den Ansichten erzeugen, die die physischen Begebenheiten reflektiert.

C.2 Elastische Graphenanpassung


Die elastische Graphenanpassung beschreibt ein Objekt also zum einen in Bezug auf
C.3 Lokale Transformationseigenschaften

Ausgehend von der elastischen Graphenanpassung ergeben sich die folgenden Fragen.

1. Können die Transformationseigenschaften aus Beispielen extrahiert werden ohne detailliertes Wissen über das konkrete Objekt?

2. Können die so gewonnen Erkenntnisse eingesetzt werden, um das Korrespondenzproblem zu lösen?

3. Welche Informationen über die physische Welt enthalten die extrahierten Transformationseigenschaften?

C.3.1 Die Statistik von Transformationen

Durch geeignete Repräsentation der Korrespondenzabbildungen zwischen einem gegeben Modellgraphen und ca. 1000 Testbildern wurden mit Hilfe der Hauptkomponentenanalyse typische Transformationen von Gesichtern ermittelt.

in diesem Sinne als Rechtfertigung für den ursprünglichen Ansatz der elastischen Graphenanpassung angesehen werden. Zusätzlich zur Reproduktion dieser Transformationen, wie sie auch bisher verwendet wurden, sind auch solche Transformation gefunden worden, die mit der dreidimensionalen Struktur von Gesichtern verknüpft sind.

C.3.2 Erkennung

Schränkt man den Raum aller möglichen Abbildungen zwischen dem Modellgraphen und dem Bild auf den linearen Unterraum ein, der die durch die Hauptkomponentenanalyse gefundenen typischen Transformationen enthält, so lässt sich die Komplexität des Korrespondenzproblems erheblich reduzieren.

Falls die gefunden Transformationen tatsächlich typisch für Gesichter sind und nicht nur für die verwendete Datenbasis, kann durch die Beschränkung auf den linearen Unterraum die Anzahl der Freiheitsgrade der möglichen Korrespondenzabbildungen eingeschränkt werden. Die daraus resultierende Reduktion der Vieldeutigkeit sollte die Qualität der Lösung des Korrespondenzproblems verbessern.

Um dies zu überprüfen, wurden Gesichtserkennungsexperimente mit der eingeschränkten Menge Korrespondenzabbildungen auf einer neuen Gesichtsdatenbasis durchgeführt. Hierbei zeigte sich, dass die so gewonnenen Korrespondenzen besser zur Erkennung geeignet sind als unbeschränkte Korrespondenzen oder solche Korrespondenzen, die durch eine von Hand vorgenommene Beschränkung erzielt wurden. Insbesondere für in der Tiefe gedrehte Gesichter erwiesen sich die durch die Hauptkomponentenanalyse gefundenen Transformationen als überlegen.

C.3.3 Punktverfolgung


Zum einen wurde die Korrespondenz zwischen zwei Punkten in aufeinanderfolgenden Bildern der Sequenz unabhängig von einander geschätzt, und die Randbedingungen wurden erzwungen durch Projektion der resultierenden Gesamtabbildung auf den linearen Unterraum. Zum anderen wurde eine Parametrisierung des Unterraums genutzt, um die Korrespondenzabbildung direkt in dem gegeben Unterraum zu schätzen.

Die direkte Schätzung im begrenzten Abbildungsraum hat sich dabei als robuster erwiesen, da lokale Vieldeutigkeiten einiger Punkte sich nicht ungehindert auf das Gesamtresultat auswirken.
C.3.4 Interpretation von Transformationen

Um zu sehen, ob die gefundenen Transformationen über die reine Beschreibung von Bildinhalten hinaus auch Information über die physischen Eigenschaften der gezeigten Objekte enthalten, wurde versucht, den gefundenen Basisvektoren des Abbildungsraums Bedeutung zuzuweisen. Die vier ersten Basisvektoren, die sich aus der Hauptkomponentenanalyse ergeben haben, schienen, die geometrischen Transformationen Translation und Skalierung zu beschreiben. Dies wurde im wesentlichen bestätigt durch einen Vergleich mit idealisierten Versionen dieser Abbildungstransformationen. Es ist allerdings nicht überraschend, dass diese Transformationen, die sich in der Bildebene abspielen und die relativ einfache Transformationen auf dem Bildinhalt darstellen, gut gefunden wurden.


C.4 Anwendungen

Im folgenden wurde anhand von zwei Beispielanwendungen gezeigt, wie die zuvor entwickelten Methoden zur Erzeugung und Integration von linearen Transformationsräumen im Rahmen komplexerer Aufgabenstellungen eingesetzt werden können.

Dabei kann die erste Anwendung auch als Motivation für den ursprünglichen Ansatz zur Ermittlung von Zwangsbedingungen aus der Statistik von Korrespondenzen angesehen werden.


C.4.1 Detailliertes Finden von Landmarken

Das gestellte Problem war, mit hoher Präzision so genannte Landmarken, d.h. vorher definierte ausgezeichnete Punkte, wie z.B. Nase oder Auge, in hochauflösenden Gesichtsbildern zu lokalisieren. Zu diesem Zweck wurde eine hierarchische Graphenanpassung
entwickelt, wobei die Landmarken in Teilbereiche zusammengefasst wurden. Für jeden dieser Teilgraphen wurden dann anhand von Beispielabbildungen, die typischen Korrespondenzabbildungen ermittelt, und wie zuvor in die Graphenanpassung integriert. So war es möglich, die Landmarken mit sehr hoher Präzision zu finden. Es zeigte sich, dass die Beschränkung auf lineare Abbildungsunterräume um so effektiver eingesetzt werden kann je mehr Freiheitsgrade, d.h. Knoten, ein Teilgraph besitzt.

C.4.2 Imitationslernen

In diesem Abschnitt geht es um die Verfolgung von Fingerspitzen durch eine Sequenz von Stereobildpaaren. Diese spezielle Aufgabe war Teil eines größeren Projektes, dass einen Roboter in die Lage versetzen sollte, die Handlungen eines Lehrers zu beobachten, um sie im Nachhinein zu imitieren und so anhand von Beispielen zu lernen. In der hier gewählten Anwendung sollte der Roboter sehen, wie eine Objekt von einem menschlichen Lehrer gegriffen wurde, um daraufhin durch Imitation der beobachteten Bewegung das Objekt selbständig greifen zu können.


C.5 Repräsentation durch Wahrnehmungsmannigfaltigkeiten

In den vorangegangenen Kapiteln wurde gezeigt, wie lineare Näherungen der Transformationseigenschaften dreidimensionaler Objekte gewonnen werden und wie diese erfolgreich bei der Lösung einer Reihe von Aufgaben eingesetzt werden können. Es ist aber ebenso klar geworden, dass es mit Hilfe einer linearen Beschreibung nicht ohne weiteres möglich ist, die Menge aller Ansichten eines Objektes topologieerhaltend zu parametrisieren.

Bevor im folgenden ein Modell vorgeschlagen wird, mit dessen Hilfe die Ansichten eines Objektes topologieerhaltend repräsentiert werden können, ist es instruktiv sich im allgemeinen zu überlegen, welcher Nachbarschaftsstruktur die Ansichten eines starren Objektes unterliegen und in welcher Beziehung diese Struktur zu den physikalischer Zuständen des Objektes steht. Es können drei Sorten von Objekten unterschieden werden.

1. **Invertierbare Parameterabbildungen**: Im einfachsten aller Fälle ist jede Ansicht eines Objektes in eindeutiger Weise einer Orientierung zugeordnet. Falls die
Abbildung vom Parameterraum in den Raum der Ansichten stetig ist, so ist auch die Umkehrabbildung stetig, und die Ansichten bilden eine Mannigfaltigkeit im Ansichtenraum.

2. **Auf Parameterteilräumen invertierbare Parameterabbildungen:** Im Falle symmetrischer Objekte, wie z.B. ein untexturierter Würfel, existieren Ansichten, die obwohl identisch, zu unterschiedlichen Parametersätzen gehören. In diesem Falle ist die Umkehrabbildung von den Ansichten auf die Parameter nicht eindeutig. Da die Vielfälsigkeit aber in gleicher Weise für alle Ansichten vorhanden ist, existiert ein Teilraum des Parameterraums auf dem die Zuordnung zu den Ansichten eindeutig und umkehrbar ist. Die Abbildung bildet also auch hier eine Mannigfaltigkeit im Ansichtenraum.

3. **Teilweise invertierbare Parameterabbildung:** Die Situation ändert sich, wenn nur eine lokale Symmetrie besteht, d.h. in diesem Zusammenhang, wenn zwei Ansichten identisch sind, ohne dass sich auch die Nachbaransichten gleichen. Ein Beispiel hierfür ist ein Würfel mit zwei gleich texturierten Seiten und vier unterschiedlichen Seiten. In diesem Falle bilden die Ansichten keine Mannigfaltigkeit, weil die Struktur im Ansichtsraum sich selbst schneidet. An diesen Schnittpunkten ist keine stetige Abbildung auf die physikalischen Parameter möglich. Solche Schnittpunkte stellen so genannte *virtuelle Übergänge* dar, weil zwei Ansichten benachbart scheinen, die eigentlich nicht direkt in einander überführbar sind. In diesem Falle kann die Struktur im Ansichtenraum aber immer noch in mehrere Mannigfaltigkeiten unter Ausschluss der Schnittpunkte zerlegt werden.

Um diesen Effekten vorzubeugen und ihre Auswirkungen zu beschränken sowie um zuvor entwickelten Verfahren zur Bildung linearer Modell Weiterhin einsetzen zu können, wird im folgenden ein Ansatz beschrieben, der es erlaubt, zunächst lokale Parametrisierungen zu finden, um dann ihren Gültigkeitsbereich sukzessiv zu erweitern.

**C.6 Aspekte**

Um eine Menge von Ansichten mit lokalen Parametrisierungen bzw. Topologien zu versehen, muss in zwei wesentlichen Punkten über das bisher Erreichte hinausgegangen werden. Zum einen steht nun keine initiale Objektrepräsentation, wie im Beispiel der Gesichter der Bündelgraph, mehr zur Verfügung. Zum anderen waren die Beispiele zuvor so gewählt, dass eine lineare Näherung in der Lage war, den gesamten Datensatz hinreichend zu parametrisieren. Dies wird im folgenden, wo wir uns mit der Struktur aller möglichen Ansichten beschäftigen wollen, sicherlich nicht mehr der Fall sein. Es ist daher nötig, sowohl die initiale Modellbildung als auch die Untergliederung in linear beschreibbare Untermengen der Ansichten zu automatisieren.
APPENDIX C. ZUSAMMENFASSUNG IN DEUTSCHER SPRACHE

Zu diesem Zweck wird mit Hilfe von datengetriebener Segmentierung zunächst eine Repräsentation in Form eines Modellgraphen für jede Ansicht erzeugt. Die Modellgraphen erlauben nun, die Ansichten miteinander zu vergleichen und ähnliche Ansichten zu sogenannten Aspekten zusammenzufassen. Es wurde gezeigt, dass die Ähnlichkeit, wie sie durch die Modellgraphen und die elastische Graphenanpassung gegeben ist, tatsächlich geeignet ist, solche Mengen von Ansichten zu finden, die sich durch eine lineare Näherung der typischen Transformationen parametrisieren lassen. Weiterhin war es möglich, die lokale Dimensionalität der Aspekte abzuschätzen, d.h. die Anzahl der Freiheitsgrade, entlang derer sich die Ansichten eines Aspektes ineinander überführen lassen, kann aus der Eigenwertverteilung der Hauptkomponentenanalyse zuverlässig geschätzt werden.

Es ist also gelungen, eine Menge von Ansichten so in Teilmengen zu zerlegen, dass den Ansichten in jeder dieser Teilmengen eine topologische Ordnung gegeben werden konnte, die die Nachbarschaft in Bezug auf die Parameter korrekt wiedergibt, die nicht direkt mit dem Aussehen sondern mit dem Zustand des Objektes in der dreidimensionalen Welt zusammenhängen. Dabei wurden die einzelnen Teilmengen unabhängig von einander behandelt, d.h. dass keinerlei Relation zwischen den Teiltopologien besteht.

C.7 Ansichtsmannigfaltigkeiten

Die Unabhängigkeit der Teilrepräsentationen hat sowohl große Vorteile als auch Nachteile. Auf der einen Seite muss auf diese Art und Weise kein Bezug auf einen einbettenden Repräsentationsraum für alle Ansichten genommen werden. Solche globalen Repräsentationsräume leiden darunter, dass die verschiedenen Ansichten eines Objektes kaum Gemeinsamkeiten haben müssen. Um aber dennoch alle Ansichten repräsentieren zu können, sind diese Räume i.a. sehr hochdimensional und zum anderen ungeeignet für alle Aufgaben, die über die reine Repräsentation hinausgehen wie z.B. das Erkennen von Ansichten. Dass das im folgenden erläuterte Verfahren auf einen solchen einbettenden Raum verzichten kann, stellt einen fundamentalen Unterschied zu allen anderen üblichen Verfahren dar, die sich mit der Erzeugung von Topologien befassen, und erlaubt zu jedem Zeitpunkt in niederdimensionalen und für die Anwendung optimierten Räumen zu arbeiten.

Auf der anderen Seite fehlt bisher jeder Zusammenhang zwischen den Teilrepräsentationen, der es erlauben würde, eine globale Topologie auf allen Ansichten zu etablieren. Um dieses Problem zu lösen, kann man sich nun den Umstand zunutze machen, dass Ansichten existieren, die in mehr als einem Aspekt repräsentiert sind. Mit Hilfe dieser Ansichten können Abstände zwischen den einzelnen Aspekten geschätzt werden. Das Verfahren der multidimensionalen Skalierung erlaubt es dann, den Aspektmittelpunkten Koordinaten im euklidischen Raum so zuzuweisen, dass die geschätzten Abstände gut wiedergegeben werden. Dieser so gefundene euklidische Raum stellt nun einen einbettenden Parameterraum dar, in dem die lokalen Topologien zur einer globalen zusammengefügt werden können. Um
C.8. DISKUSSION UND AUSBLICK


Um zu zeigen, dass die so erzeugte Topologie Eigenschaften des den Ansichten zugrundeliegenden Objektes widerspiegelt, wurde sie zur Grundlage eines linearen Interpolationsalgorithmus gemacht. Mit dessen Hilfe war es möglich, den Ansichten unter Verwendung weniger Stützstellen Winkel zuzuordnen, die die relative Position des Objektes zur Kamera beschrieben. Es konnte gezeigt werden, dass die so gewonnene Posenschätzung eine Genauigkeit erreicht, die vergleichbar ist mit Verfahren, die die Topologie bereits voraussetzen. Nur an solchen Stellen traten Probleme auf, an denen die Selbstähnlichkeit des Objektes zu den oben besprochenen virtuellen Übergängen führte. Die so entstandenen Fehler in der Interpolation waren aber auf kleine Bereiche beschränkt und erstreckten sich nicht auf die gesamte Objektrepräsentation.

C.8 Diskussion und Ausblick

Es wurde gezeigt, dass es möglich ist, eine Objektrepräsentation nur aus einzelnen ungeordneten Ansichten zu erzeugen. Der verfolgte Ansatz erlaubte es, Ansichten zu parametrierten Aspektrepräsentationen zusammenzufügen und eine globale Struktur auf den Aspekten zu errichten. Dieses hierarchische Vorgehen stellte mit jedem Schritt neue Funktionalität zur Verfügung und baute auf ihr auf.

So ermöglichte die Repräsentation in Form von Modellgraphen die Erkennung bereits bekannter Ansichten auch unter leichten Variationen. Durch das Zusammenfügen von Ansichten zu Aspekten konnten lineare Näherungen an die Transformationseigenschaften von Objekten gewonnen werden, die wiederum zur einer verbesserten Lösung des Korrespondenzproblems beitragen und eine topologische Ordnung auf den Ansichten erzeugten. Abschließend wurde gezeigt, dass sich alle Ansichten zu einer topologieerhaltenden Objektrepräsentation zusammenfügen lassen, deren Struktur ein schnelles Lernen in Bezug auf die dreidimensionalen Eigenschaften von Objekten erlaubt. Im Falle von Objekten, die Selbstähnlichkeiten enthielten, so dass sie sich nicht Form einer einzigen Mannigfaltigkeit beschreiben ließen, wurde gezeigt, dass sich die so entstehenden Probleme auf kleine Bereiche der Repräsentation beschränken lassen.

Darüberhinaus ermöglichte das Zusammenfassen von Ansichten zu lokalen Aspekten eine globale Parametrisierung der Objekttransformationen, ohne Bezug auf einen einbetenden Raum der Ansichten zu nehmen, wie es z.B. bei künstlichen neuronalen Netzen und verwandten Methoden nötig ist. Dadurch konnten die verschiedenen Repräsentationen zu
jedem Zeitpunkt niederdimensional konstruiert werden.


BIBLIOGRAPHY


