Object Recognition with Representations based on Sparsified Gabor Wavelets used as Local Line Detectors

Norbert Krüger
Institut für Informatik und praktischer Mathematik
Christian-Albrechts-Universität zu Kiel
Preußerstrasse 1-9, 24105 Kiel
Germany
nkr@ks.informatik.uni-kiel.de

Abstract. We introduce an object recognition system (called ORASSYLL) in which objects are represented as a sparse and spatially organized set of local (bent) line segments. The line segments correspond to binarized Gabor wavelets or banana wavelets, which are bent and stretched Gabor wavelets. These features can be metrically organized, the metric enables an efficient learning of object representations. Learning can be performed autonomously by utilizing motor–controlled feedback. The learned representation are used for fast and efficient localization and discrimination of objects in complex scenes.
ORASSYLL has been heavily influenced by an older and well known vision system [4, 9], and has also been influenced by Biederman's comments to this older system [1]. A comparison of ORASSYLL and the older system, including some remarks about the specific role of Gabor wavelets within ORASSYLL, is given at the end of the paper.

1 Introduction

In this paper we describe a novel object recognition system called ORASSYLL (**O**bject **R**ecognition with **A**utonomously learned and **S**parse **SY**mbolic representations based on **L**ocal **L**ine detectors). In ORASSYLL representations of object classes can be learned autonomously. The learned representations are used for a fast and efficient localization and identification of objects in complicated scenes.

We facilitate and guide learning by carefully selected a priori knowledge. Important constraints are the restriction of features to localized (bent) line segments (PF1), their metric organization (PF2), their hierarchical processing (PF3) and the sparse representations of objects by these features (see figure 1). Other contraints, discussed in detail in [3], are concerned with the division of the feature space in independent subspaces (PL1: Independence), its temporal organization (PL2: Correspondence) and statistical criteria for the evaluation of significant features for an object class (Invariance Maximization (PE1) and Redundancy Reduction (PE2)).

A cruical aspect of our work and an important difference to an older system [4, 9] is the specific application of Gabor wavelets as expressed in the constraints

PF1, PF2 and PF4. PF1 states that we apply Gabor wavelets as local line detectors, i.e., we assign a symbolic meaning to them. This meaning also enables the imbedding of Gabor wavelets into a metrically organized space (PF2) as well as for a sparse object representation (PF4).

Our representation of a certain view of an object class comprises only important features, learned from different examples (see figure 2 left). In section 2 we formalize PF1 by assigning a local line segment to Gabor wavelets or banana wavelets respectively (see figure 1a,b). In addition to the parameters frequency and orientation banana wavelets possess the properties curvature and elongation. The space of Gabor or banana wavelet reponses is very large. An object can be represented as a configuration of a few of these features, therefore it can be coded sparsely (PF4) (see figure 1c). The feature space can be understood as a metric space (PF2), its metric representing the similarity of features. This metric is essential for feature extraction and the learning algorithm (section 3). The banana wavelet responses can be derived from Gabor wavelet responses by hierarchical processing (PF3) to gain speed and reduce memory requirements. The sparse representation combined with the hierarchical feature processing allows a fast and effective locating (section 4).

In order to avoid the necessity of manual intervention for the generation of ground truth we equip the system with a mechanism which can produce controlled training data by moving an object with a robot arm and following the object by fixating the robot hand. The robot produces training data on which a certain view of an object is shown with varying background and illumination but with corresponding landmarks having the same pixel position in the image (see figure 2 left). We apply the learning algorithm to this data to extract an object representation (see figure 2 left,v). Another way to avoid manual intervention is one–shot learning (see figure 2 left), which already allows for the extraction of representations successfully applicable to difficult discrimination tasks.

ORASSYLL has been heavily influenced by an older and well known vision system [4, 9], and has also been influenced by Biederman's comments to this older system [1]. A comparison of ORASSYLL and the older system, including some remarks about the specific role of Gabor wavelets within ORASSYLL, is given in section 6.

2 The Feature Space

The principle PF1 gives us a significant reduction of the search space. Instead of allowing, e.g., all linear filters as possible features, we restrict ourself to a small

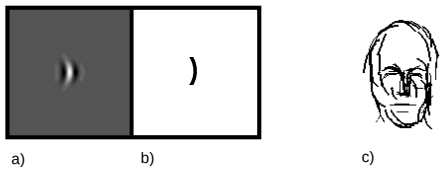

Fig. 1. a: Arbitrary wavelet. b: Corresponding path. c: Visualization of a representation of an object class. Gabor or Banana wavelets with lower frequencies are represented by line segments with larger width.

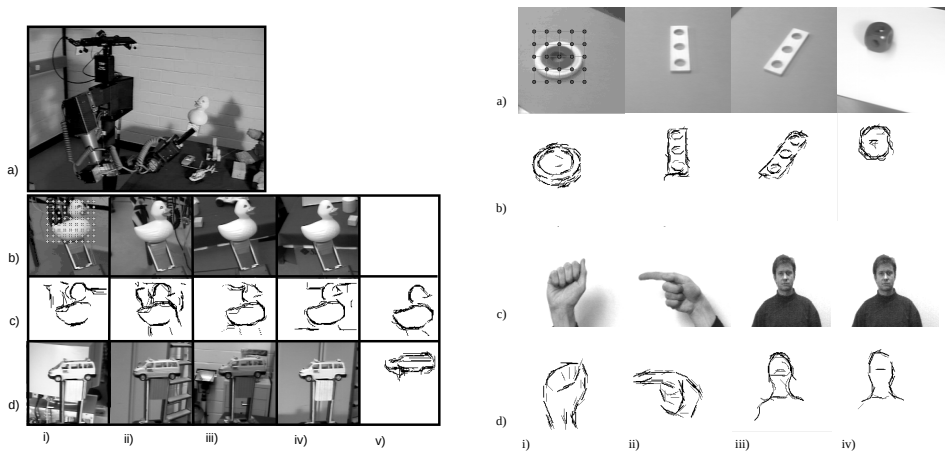

Fig. 2. Left: Autonomous learning: a) The robot arm with the camera. **b)** The "retinal" images produced by following the robot arm holding a toy–duck. **c,i–iv)** Significant Features per Instance extracted in an rectangular region (shown in b,i). **c,v)** Learned representation. **d)** Training data and learned representation for a toy car. **Right: One–shot learning:** Row a) and c) show the objects to be learned in front of homogeneous backgound. Row b) and d) show the extracted representations. For all objects a rectangular grid was roughly positioned on the object as in the first image a,i).

subset. Considering the risk of a wrong feature selection it is necessary to give good reasons for our decision. We argue that nearly any 2D–view of an object can be composed of localized curved lines. Furthermore, the fact that humans can easiliy handle line drawings of objects strengthens our assumption PF1.

Gabor and Banana Wavelets: A banana wavelet $B^{\mathbf{b}}$ is a complex–valued function, parameterized by a vector $\mathbf{b}$ of four variables $\mathbf{b} = (f, \alpha, c, s)$ expressing the attributes frequency (f), orientation (α), curvature (c) and size (s). It can be understood as a product of a rotated (and curved) complex wave function $F^{\mathbf{b}}$ and a stretched two–dimensional Gaussian $G^{\mathbf{b}}$ rotated (and bent) according to $F^{\mathbf{b}}$ (see figure 1a).

Our basic feature is the magnitude of the filter response extracted by a convolution with an image. A Gabor or banana wavelet $B^{\mathbf{b}}$ causes a strong response at pixel position $\mathbf{x}$ when the local structure of the image at that pixel position is similar to $B^{\mathbf{b}}$ (see [3]).

The Feature Space: The six–dimensional space of vectors $\mathbf{c} = (\mathbf{x}, \mathbf{b})$ is called the *feature space* with $\mathbf{c}$ representing the banana wavelet $B^{\mathbf{b}}$ with its center at pixel position $\mathbf{x}$ in an image. In [3] we define a metric $d(\mathbf{c}_1, \mathbf{c}_2)$. Two coordinates $\mathbf{c}_1, \mathbf{c}_2$ are expected to have a small distance d when their corresponding kernels are similar, i.e., they represent similar features (PF2).

Approximation of Banana Wavelets by Gabor Wavelets: To reduce computational requirements for the extraction of the large feature space we have defined an algorithm to approximate banana wavelets from Gabor wavelets and banana wavelet responses from Gabor wavelet responses (see [3]). By this hierarchical processing (PF3) we achieve a speed up to a factor 5.

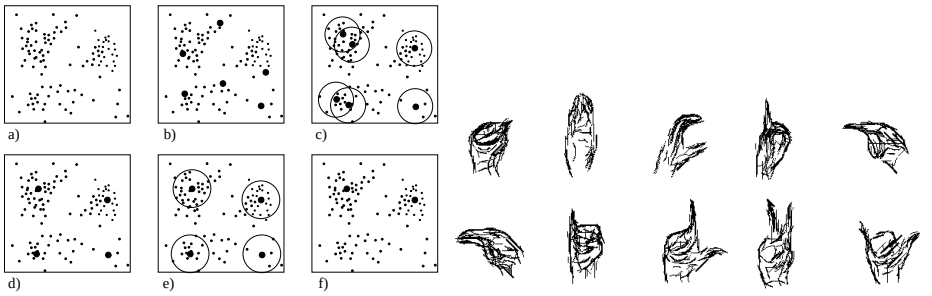

Fig. 3. Left: Clustering: a) Distribution of the significant features per instance extracted at a certain landmark. b) Codebook initialization. c) Codebook vectors after learning. d) Substituting sets of codebook vectors with small distance by their center of gravity. e) Counting the number of elements within a certain radius. f) Deleting codebook vectors representing insignificant features. **Right: Learned Representations of 10 different hand postures** The manually provided ground truth consists of 6 pictures per hand posture with a grid consisting of approximately 40 landmarks is placed.

3 Learning

Extracting Significant Features Per Instance: Our aim is to extract the local structure in an image I in terms of (curved) line segments expressed by Gabor or banana wavelets. We define a *significant feature per instance* of an object by two qualities. Firstly it has to cause a strong response, secondly it has to represent a maximum within a local area of the feature space. Figure 2 right b,c) and left c,i–iv) show the significant features per instance for some objects (each wavelet is described by a curve with same orientation, curvature and size).
One–shot learning: By positioning a rectangular grid on a roughly segmented object (see figure 2 left a,i) in front of homogeneous background and extracting significant features per instance as described above suitable representations of objects can already be extracted. These representations are successfully applied to difficult discrimination tasks.
Clustering: After extracting the significant features per instance in different pictures we apply an algorithm to extract invariant local features for a *class of objects*. Here the task is the selection of the *relevant features* for the object class from the noisy features extracted from our training examples (see figure 2 left c,i–iv). We assume the correspondence problem to be solved, i.e., we assume the position of certain landmarks of an object to be known on pictures of different examples of these objects. In some of our simulations we determined corresponding landmarks manually, for the rest we replaced this manual intervention by motor controlled feedback (see section 5).

In a nutshell the learning algorithm works as follows (illustrated for two dimensions in figure 3 left): a–c) For each landmark we express the significant features per instance of all training examples by six dimensional codebook vector $(\mathbf{x}, \mathbf{b})$, representing the pixel position and the parameter frequency, orientation, curvature and elongation. We optimize the codebook vectors by the LBG vector quantization algorithm [5]. d) Codebook vectors with small distances are

substituted by their center of gravity (PE2: reduction of redundancy). e,f) A significant feature for an object class is defined as a codebook vector expressing many data points. That means the feature corresponding to the code book vector or a similar feature (according to our metric d) often occurs in our training set, i.e., has high invariance (PE1). We end up with a graph with its nodes labeled with banana wavelets representing the learned significant features (see figure 2 left dv, ev). The edges of the graph labeled with metric relations of the landmarks.

4 Matching

To use our learned representation for location and classification of objects we define a similarity function between a graph labeled with the learned banana wavelets and a certain position in the image. A *total similarity* averages *local similarities*. The local similarity expresses the system's confidence whether a pixel in the image represents a certain line segment. The graph is adapted in position and scale by optimizing the total similarity. The graph with the highest similarity determines the size and position of the objects within the image.

In a nutshell the local similarity is defined as follows (for details see [3]): The magnitude of the filter responses depends significantly on the strength of edges in the image. However, here we are only interested in the presence and not in the strength of edges. Thus, in a second step a function $N(\ \ )$ normalizes the real valued filter responses into the interval $[0, 1]$. The value $N(\mathbf{c})$ represents the likelihood of the presence or absence of a local line segment corresponding to $\mathbf{c} = (\mathbf{x}, \mathbf{b})$. This normalization is based on the "Above Average Criterion":

AAC a line segment corresponding to the banana wavelet $\mathbf{c}$ is present if the corresponding banana wavelet response is distinctly above the average response.

For a learned feature and pixel position in the image we simply check whether the corresponding banana response is high or low, i.e., we look at the normalized wavelet response $N(\mathbf{c})$. The total similarity (which is optimized during matching) is simply the average over all these local confidences. Because of the sparseness (PF4) of our representation only a few of these checks have to be made, therefore the matching is fast. Because we make use only of the important features, the matching is efficient.

5 Simulations

Learning of Representation: Firstly we apply the learning algorithm to data consisting of manually provided landmarks. Our training sets consist of a set of approximately 60 examples of an object viewed in a certain pose. As objects we use cans, faces, and hand postures. Corresponding landmarks are defined manually on the different representatives of a class of objects for the learned representation for hand postures.

To avoid the manual generation of ground truth we can either apply one–shot learning (see section 3) or make use of motor controlled feedback: By moving an object with a robot arm and following the object by keeping fixation relative to the robot hand using its known 3D position, we produce training data in which a certain view of an object is shown with varying background and illumination

	Repres.		Trafo		Performance	
	nb. reps	rep	approx	sec.	sec. match	Recog.
1)	10	standard	no approx	17.0	9.5	93 %
2)	10	one instance	approx	4.9	12.4	80 %
3)	10	bunch graph		0.9	18.0	93 %
4)	10	standard	no approx	17.0	9.5	90 %
5)	10	standard	approx	4.9	9.5	80 %
6)	10	bunch graph		0.9	18.0	65 %

Table 1. Matching results for hand posture recognition (for interpretation see text).

but with corresponding landmarks in the same pixel position within the image (see fig 2 left b,d). Then we can apply our learning algorithm with a rectangular grid roughly positioned on the object (see figure 2 left b,i). For the generation of ground truth for frontal faces we recorded a sequence of pictures in which a person is sitting fixed on a chair. Illumination and background is changed as for cans. To extract representations for different scales we apply the learning algorithm to the very same pictures of the different sequences scaled accordingly.
Matching: For the problem of face finding in complex scenes with large size variation a significant improvement in terms of performance and speed compared to the older system [4, 9] could be achieved. In [6] face recognition with binarized banana wavelets was performed on a very large data set (more than 700 pictures) with size variation of faces between 40 and 60 pixel, inhomogeneous background and uncontrolled illumination. For this set performance was 95%.

Our test sets of hand postures contain images of 10 different hand postures (figure 3 (right) shows the learned representations) in front of homogeneous background with controlled illumination (row 1–3, 240 images) and with a second set containing images with inhomogeneous background and varying illumination (row 4–6, 200 images). Matching with ten representations (one for each hand posture) takes 9.5 seconds and recognition rate for the first set was 93% (first row). The simulations corresponding to the second row were performed with representations extracted by one-shot learning. The performance is still remarkably high (80%). The performance with the bunch graph approach as described in [8] is given in the third row. Results for the test set with uncontrolled background and illumination is shown in row 4–6. For the first test set performance within the bunch graph approach [9, 8] is comparable to ORASSYLL. For the second and more difficult, set performance of ORASSYLL is significantly better.

6 Comparison with the Jet–based System

ORASSYLL has been heavily influenced by an older and well known vision system [4, 9, 8] in the following called jet–based system, and has also been influenced by Biederman's comments to this older jet–based system [1]. The system [4, 9] was very successfully applied to face recognition. High correlation between the system's and human's face recognition performance has been shown [1]. However, Biederman and his associates [1] also have shown that the system [4, 9] has only low correlation to human object recognition.

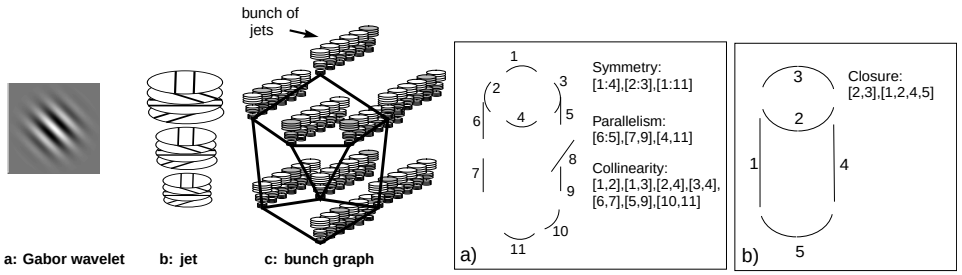

Fig. 4. Left: Representation of objects within the jet–based system: a: a Gabor wavelet (real part). **b**: a jet calculated as a set of Gabor wavelets (the discs symbolize the different frequencies and directions of **k**). **c**: a bunch graph. **Right: Gestalt principles as low order relations of local line segments: a:** Sparse representation of a can with local curved lines and lists of second order Gestalt relations between. **b:** Smoothing and grouping of the representation in a).

As models for objects the older system also employs labeled graphs. The edges of graphs are labeled with distance vectors between node positions. Nodes are labeled with jets [4] or bunches of jets [9] respectively. In a bunch of jets each jet is derived from the image of a different example of the view of an object. A is bunch thus covering a variety of forms a single landmark may take. This structure is called *bunch graph* [9].

Jets are derived from a set of linear filter operations in the form of convolutions of the image $I(\mathbf{x})$ with a set of Gabor wavelets, of different wavelength and orientation (see figure 4 left a). A jet is formed by the set of complex values rendered by all wavelets centered at a given position of the image (see figure 4 left b). Due to the spatial extent of the wavelets, jets describe a local area around their position. A bunch $\mathcal{B}$ of jets taken at the same landmark (that is, at corresponding positions) of different examples of a certain view of an object class forms a generalized representation of this landmark (see figure 4 left c).

Jet components a_j (the index j standing for length and orientation of the components' wave vectors) are the magnitude a_j of Gabor wavelet reponses. The similarity between two jets $\mathcal{J}$ and $\mathcal{J}'$ is defined as the normalized scalar product of the two jets:

$$S(\mathcal{J}, \mathcal{J}') = \frac{1}{\sqrt{\sum_j a_j^2 \sum_j a_j'^2}} \cdot \sum_j a_j a_j' \tag{1}$$

Conceptional Differences of Object Representations in the Jet–based System and ORASSYLL: The object representation on ORASSYLL shows four conceptional (D1–D4) differences to the representation based on jets and bunches of jets.

- **(D1)** The distinction curvature vs. straightness can be explicitly used as a feature.
- **(D2)** A restriction to a specific set of binary features of high complexity (more precisely, local (curved) line segments) is imposed for object representation.

- **(D3)** The object representations are sparse.
- **(D4)** A metric is utilized as an additional structure of the feature space.

In the next subsection we will discuss how these differences influence the recognition process.

6.1 Comparison with Jets and Bunches of Jets: Two Arguments in Favor of ORASSYLL for the task 'object recognition'

In this section two advantages of the object representation within ORASSYLL are discussed.

First Argument: In a Jet Significant and Insignificant Features are Lumped together whereas in ORASSYLL Features are Stored Separately. Therefore Matching and Extraction of Significant Features is Facilitated: In a jet, significant and insignificant features are lumped together. Even when a single Gabor wavelet response gives information about the occurrence of a local line with a certain orientation, a jet always represents the whole local image patch. The jet similarity (1) reflects the relative strengths of a complete set of Gabor wavelet responses at the actual pixel position, and therefore reflects the fit to a whole local region. For example, a local area of an object may have an edge with a certain orientation resulting in a strong response of the corresponding Gabor wavelet. The occurrence of an edge with different orientation in the background causes a strong response for the Gabor wavelet with different orientation. Because the denominator in equation (1) increases by the "background–response", the relative strength of Gabor wavelet responses, and therefore the similarity (1) changes. In [3] it has been demonstrated that — because the relative strength of Gabor wavelet responses varies significantly with changes of background and illumination — the similarity (1) is more sensitive to these sources of noise compared to the similarity function applied within ORASSYLL.

In ORASSYLL two different distance functions for learning and matching are used: Firstly, the metric defines a distance between features (D4). For learning, features at a close distance are grouped together within one cluster but features at a large distance are treated as *separate*. The metric of the feature space reflects the difference of properties of features such as difference in space, curvature or orientation. This allows to distinguish between significant features and insignificant features (e.g., corresponding to the background) and to keep only the significant features within the object representation (see figure 2 left). Secondly, the similarity of a binarized banana wavelet to a local image patch (based on the Above Average Criterion) indicates the presence or absence of the learned feature fairly independent of background and illumination (as shown in [3]) and allows for a comparison of only the learned and significant features to the image.

Second Argument: Coding and Detection of Important Relations such as Collinearity, Parallelism and Symmetry is more Difficult with Jets than with Binarized Banana Wavelets: An important issue within ORASSYLL is the definition of a local criterion for the presence of a local (curved) line. Maybe there does not even exist such a completely satisfying *local* criterion and the presence of a local line segment depends on the *context*. Compared to the older system, invariance to changes in illumination and background is

significantly higher, but still, variation occurs as it has been demonstrated in [3].

In [2] within the framework of ORASSYLL, it could been shown that collinearity and parallelism can be detected and mathematically characterized in natural images with binarized Gabor responses. Without binarization, i.e., without a transformation to a feature of higher complexity corresponding to a localized line (D2), these two Gestalt principles are barely detectable in the statistics of natural images.

A sparse representation of objects (D3) allows for the description and detection of Gestalt principles as low–order statistics of feature relations. (see figure 4 right) We suggest that this coding also facilitates the reliable recognition of a local line segment by integrating contextual information because interactions between features such as inhibition and reinforcement can be defined according to Gestalt principles within the feature space. Coding of Gestalt principles with jets would presuppose higher order statistics. Therefore learning and integration of Gestalt principles becomes more difficult within the jet approach.

Acknowledgment: I would like to thank Christoph von der Malsburg, Gabriele Peters, Laurenz Wiskott and Michael Pötzsch for fruitful discussion.

References

1. I. Biederman and P. Kalocsai. Neurocomputational bases of object and face recognition. *Philosophical Transactions of the Royal Society: Biological Sciences*, 352:1203–1219, 1997.
2. N. Krüger. Collinearity and parallism are statistically significant second order relations of complex cell responses. *Neural Processing Letters*, 8(2), 1998.
3. N. Krüger. *Visual Learning with a priori Constraints (Phd Thesis)*. Shaker Verlag, Germany, 1998.
4. M. Lades, J.C. Vorbrüggen, J. Buhmann, J. Lange, C. von der Malsburg, R.P. Würtz, and W. Konen. Distortion invariant object recognition in the dynamik link architecture. *IEEE Transactions on Computers*, 42(3):300–311, 1992.
5. Y. Linde, A. Buzo, and R.M. Gray. An algorithm for vector quantizer design. *IEEE Transactions on communication*, vol. COM-28:84–95, 1980.
6. H.S. Loos, B. Fritzke, and C. von der Malsburg. Positionsvorhersage von bewegten objekten in groformatigen bildsequenzen. *Proceedings in Artificial Intelligence: Dynamische Perzeption*, pages 31–38, 1998.
7. Michael Pötzsch, Thomas Maurer, Laurenz Wiskott, and Christoph von der Malsburg. Reconstruction from graphs labeled with responses of gabor filters. In C. v.d. Malsburg, W. v. Seelen, J.C. Vorbrüggen, and B. Sendhoff, editors, *Proceedings of the ICANN 1996*, Springer Verlag, Berlin, Heidelberg, New York, Bochum, July 1996.
8. J. Triesch and C. von der Malsburg. Robust classification of hand postures against complex background. *Proceedings of the Second International Workshop on Automatic Face- and Gesture recognition, Vermont*, pages 170–175, 1996.
9. L. Wiskott, J.M. Fellous, N. Krüger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 775–780, 1997.

This article was processed using the LaTeX macro package with LLNCS style