# Developing Abstractions for Canonical Patterns: A Perspective

***under review with Computer Vision and Image Understanding***

*Christoph Rasche*
*Institut für Psychologie, Justus-Liebig Universität*
*Otto-Behagel-str. 10, F1*
*35394 Giessen, Germany*
*fax: +49 641 99 26 119, phone: +49 641 99 26 106, email: rasche15@gmail.com*

**Abstract**

A perspective for the construction of a fast categorization process is formulated, that proposes an exhaustive parameterization of structure and appearance. The central goal is to use the parameters to create appropriate multi-dimensional spaces that express contours, areas and groupings thereof. This is pursued in a framework consisting of a decomposition and a synthesis process: the decomposition process partitions structure into basic contour segments and areas, which then are integrated to complex descriptions in a synthesis process.

*Key words:* image classification; shape description, contour description, contour grouping, texture

## 1. Introduction

Traditional approaches to image understanding have sketched a system, that would meticulously reconstruct a scene by starting with a surface layout [1, 2] or by image segmentation [3, 4] and/or a detailed contour image [5] or a mixture of techniques [6]. In such approaches it was implied that the reconstruction process is supposed to work error-free because only then a unique assignment could be made to the corresponding category. While this perspective brought forth much elegant work and is still continued [7, 3, 4, 8], it was

so far limitedly successful at understanding the semantic content of a large number of categories (see Keselman for a review [9]). It is only in the past decade, when vision scientist tried to circumvent this detailed reconstruction process by developing systems that extract the meaning in a 'direct' (or immediate) way. We now call that the 'fast categorization' approach. There exists two prominent systems following this approach, the 'spatial envelope' system by Oliva and Torralba [10], and the object categorization system by Perona's group (e.g., [11]).

The spatial envelope is a 'holistic' scene description based on 5 perceptual dimensions (naturalness, openness, roughness, expansion and ruggedness) and allows for the categorization of super-ordinate categories such as streets, highways and coasts. Image preprocessing occurs with a modified Fourier Transform [10]. The authors state explicitly that this fast categorization process could occur without processes of image segmentation and without processes involving grouping operations, a belief inspired by the short duration of the human categorization process. The object-categorization system by Perona and co-workers is based on a mixture of methods such as the principal component analysis and orientation histogramming (e.g. [12]). The system performs on 101 sub-ordinate categories.

Both systems are good at discriminating their image classes, but once the image is classified (categorized), the process of understanding parts (components or regions) of the image requires a novel preprocessing. For instance, in case of the spatial envelope system, a preprocessing based on local orientations was developed, that allows for a visual search [13], an effort which appears to be a move toward a structural description.

Both systems attempt to emulate the 'fast categorization' process of humans in the sense that the goal is to arrive at the semantic meaning first, before a detailed reconstruction is carried out. The latter is carried in the human system by exploiting eye movements. Whereby Oliva and Torralba make an explicit connection to this process, the studies by Perona and coworkers do not express such a motivation. The fast categorization process works

only for 'canonical views', that is for scenes or objects seen from a regular or typical viewpoint in humans [14] - and does so only in those computer systems. Their image collections contain only such canonical views [10, 12]. Thus, the purpose of the fast categorization process is different from other recognition processes. For instance, some systems were developed for the view-point invariant recognition of objects ([15, 16], see Keselman for a review [9]). Fast categorization should also not be confused with object search (or localization) [17, 18, 19]. In such approaches an image is searched for a specific object, which also often appears in a canonical view, but which often covers only part of the image.

. But what has not been pursued yet is a fast categorization system that is based on a parameterization of contours and their relations. Specifically, we pursue an abstraction of contours that describes the contour geometry explicitly by parameters. We have previously introduced our approach and extend it here substantially, with the goal to open a novel perspective on the issue of fast categorization. Our perspective fundamentally distinguishes itself from the above two mentioned 'fast categorization' approaches in particular by the following reconstruction concepts:

1) Grouping is an explicit part of the reconstruction process, which is rejected by Oliva and Torralba, and not pursued by Perona's group. Thus, while we believe that some sort of gradual reconstruction takes place, it is not the type of meticulous reconstruction as pursued in the above reviewed approaches (see also [20] for arguments).

2) The preprocessing output allows to understand parts of the image as the reconstruction process is based on a description of contours and their relations.

But there is also some common ground with the approach by Oliva and Torralba, for instance see point number 2 in the perspective section.

*1.1. Perspective*

We regard a gray-scale (intensity) image as made of two types of information, structure and appearance. These are two terms that are often used in the field but may require further specification to express our perspective as precisely as possible:

- *Structure*: corresponds essentially to a (fragmented) contour image. The structure is independent of the fine/coarse scale and can comprise the entire scale space or just a slice of it (a given scale). The structure can be of any type: scene, object, texture or shape; it thus could be even a single dot on an otherwise empty background. It is best described by some sort of relation between basic contour segments (with basic we mean a curved, straight or wiggly contour segment).

- *Appearance*: corresponds to the luminance distribution along contours, e.g. contrast, and to the luminance distribution between contours (of a region), e.g. the 'fuzziness' (roughness). It is best described statistically.

Theoretically, the two types of information are not clearly distinct, as any appearance seen at a very fine scale constitutes in itself a structure again. Thus, it is an issue of scale in principle. But practically, when operating with images of limited resolution, e.g. 200x300 pixels, then this is a useful distinction as even on the finest scale there exist regions which are more meaningfully expressed as appearance statistics than as a structural relation. Our perspective on the fast categorization process rests on the following other intuitions:

1) Dominance of structure: Structure is the more important type of information than appearance. The reasoning is as follows: shapes or objects can be described as a set of features whose geometry in turn can be described in a relatively simple manner (see also [21] for arguments). A scene - consisting of several objects - is then merely a more complex shape, but typically fragmented. The contour image of a gray-scale image is not always a unique identifier of its category. But together with a parameterization of the appearance statistics, the assignment should become facile.

2) Structural association: The term structural description is typically understood as the deterministic relation of a fixed set of parts/components (palmer 99). However, the presence of structural variability literally prohibits such a rigid description for the purpose of fast categorization. Instead, the description of category representations should be structurally looser and should rather express an 'association'. Loose representation have already been proposed since early on [22, 23] and have been applied in systems pursuing a search-and-recognition strategy [24] or determining exact object pose [25]. Nelson and Selinger use the term 'Cubist' representation [17] and their metaphor comes closest to what we imagine as suitable category representations. But they did not propose specific relations or groupings amongst contours, which we consider as important. Such (global) relations should express the pattern of a canonical view as a whole; they should capture its holistic nature in some sense. This was already formulated by Oliva and Torralba for scenes and was given the term 'spatial envelope'. But we think it equally holds for any type of input (see point 6) and we therefore use the term 'canonical pattern'. It is the representation that allows the quickest access to the basic-level category.

3) Late classification: The idea of structural description is also associated with a classification of parts or components into rigid features, such as a L feature, vertex feature, even the geometry of lines (e.g. [22, 26, 27, 28, 21]). But such a classification should be avoided as it does not allow to deal well with structural variability. Instead, such a part classification should be avoided or at least occur 'late'. For that reason, features are expressed in a multi-dimensional space in which the variability appears as a subspace, and in which the features can be discriminated or compared by a simple distance measurement (see also 2.2.1b) in [20]).

4) Multiple descriptors: Ideally, there existed a single multi-dimensional space that can express all possible structural relations. Practically, this seems not possible and that is the reason why a variety of features were suggested (see citations above). Our intutition is that there exists a limited number of

spaces, which can express these features and their structural associations

5) Redundant representation: Category representations need to be highly redundant, as an image is typically so noisy that only the use of multiple structure and appearance descriptors can provide a unique category assignment. How else can one assign a low-resolution image, e.g. 32x32 pixels, to its corresponding basic-level category [29]? The very same structure can appear in one image with very characteristic contours, which in another image is so fragmented that recognition needs to rely more on the appearance parameters.

6) Reconstruction process: as mentioned in the introduction, the assumption of an error-free should be loosened. Accidental groupings are part of the categorization process. We assume that the multitude of extracted descriptors - some of which are accidental -, combined with the redundant category representations, allow for a unique assignment of the image to a basic-level category.

7) Arbitrary input: From point 1 it should have became clear, that the fast categorization process should work for any type of input: scenes, objects, shapes and texture. We repeat it here to emphasize the point. A distinction between these input types is difficult in any case: for instance, a sunset scene can consist of only a horizontal contour and a circle - plus appearance statistics for sky and water. In comparison, most shapes are more complex than the structure of such a sunset scene.

**Summary.** The central challenge of our approach is the formation of multi-dimensional spaces (point 4). Such a multi-dimensional space needs the appropriate degree of generality: too little generality means the space is incapable of dealing with structural variability; too much generality means the space is unspecific in discriminating. For instance, while the 5-dimensional space of the spatial envelope is sufficient in discriminating a limited set of super-ordinate categories, it is not sufficient to carry out finer discrimination [10]. Yet it certainly proved the representational power of a well-designed multi-dimensional space. Our approach faces particularly the challenge of

orchestrating the large number of parameters. On the one hand they can be very specific when they are used as dimensions of vectors. On the other hand, such vectors can be too specific and may not be appropriate to describe certain categories. Thus, one is confronted with the curse of dimensionality but this can be mastered by careful design and systematic testing.

## 2. Framework

Our recognition system consists of a decomposition, a synthesis and a matching process. The decomposition process partitions a structure into basic contour segments and areas (regions). Those segments are then described by geometric (structural) and appearance parameters. The synthesis process integrates those basic contours and areas to complex descriptors such as intersections and groupings representing or outline shapes or simple structures and their appearance. The decomposition and synthesis process generate a list of descriptors, which then are compared against the list of descriptors of individual category representations, called the matching process. There is no exact separation between the decomposition process and the synthesis process, as some of the transformations can also be regarded as a synthesis process, e.g. the symmetric-axis transformation.

The decomposition process has been elaborately presented in a previous publication [20], but is here summarized as we have done some minor alternations and important extensions to it. The synthesis process in particular is the novelty of this study.

### *2.1. Decomposition*

The decomposition process performs two types of transformations. One is the transformation of contours into a local/global space (LG space), the other is the transformation of structure into a symmetric-axis field. The output of those transformations is then partitioned into basic segments (contour segments and areas respectively).

*2.1.1. Contours*

The contour analysis starts with the creation of the LG space, followed by contour partitioning, followed by segment extraction and ending with contour description. The novelty in this study is the addition of another label, the straightness label, and the process of segment extraction.

. A contour is iterated with a window which classifies a segment into the two labels arc (bow) and inflexion, and which determines the amplitude of the segment. For a given window size, this leads to the 'bowness' and inflexion signature, $\beta(v)$ and $\tau(v)$ ($v$ = arc length variable). In this study, a straightness label is added, as the lack of a bowness or inflexion block is not unique enough to determine whether a segment is straight or not:

$$\gamma(v). \tag{1}$$

The straightness signature is suppressed ($\gamma$ set to 0) if at the same location a (positive) bowness value is present in the same or any higher window level ($\omega$). For a range of window sizes the resulting signatures describe the LG space, one for bows ($\beta_\omega(v)$) and one for inflexions ($\tau_\omega(v)$), and now also including the one for straightness:

$$\gamma_\omega(v). \tag{2}$$

Figure 2 shows the output of the contour decomposition in which straight and curved segments are marked as squares and circles respectively. Complete straightness is indicated by an amplitude set to a value of 0.5 (straightness label stippled), whereas a value of 0 means lack of straightness.

After creation of this space, a contour is partitioned at points of U turns, resulting in 'coarsely' elongated contours.

*Segment Extraction.* Many (partitioned) contours consist of multiple curved and straight segments and are often irregular (alternating). This irregularity can be potentially category-characteristic, as for instance the vertical wiggly contour of a person's silhouette or the horizontal contour of a landscape

scene. For that reason, further partitioning is potentially detrimental to building distinct category representations and we therefore keep such contours. But we extract large straight and curved segments as they could be locally grouped with other neighboring contours. For instance the straight segment describing the leg of a person's silhouette, could form a distinct grouping with the straight segment of the other leg. We therefore extract such segments if they are of a minimum length.

This segment extraction could occur at different local/global levels as it is a priori not clear what the appropriate level of description is, or put differently, it is a matter of context. We therefore contrived an algorithm, that extracted global segments first but that would still allow for the extraction of smaller segments. The LG space in figure 1 serves as an example: the bowness block with the largest spatial extension of the LG space is identified, see window no. 7 (block ranging from $v = 1$ to $v = 40$); the location of the block's maximal amplitude is taken as the point of highest curvature (indicated by a circle in the contour display [upper right]). This maximum block suppresses more local (but not all local) bowness blocks during subsequent extractions (the bowness blocks in windows no. 6 and 5). In a 2nd round of identification, the next wide bowness block is selected, the block in window no. 5 (ranging from $v = 33$ to $v = 59$) and its local neighborhood is suppressed. This identification and suppression procedure is repeated until all large blocks are identified. A minimum block size was set. The identified block ranges are then used to extract 3 segments from the original contour. Thus, the contour decomposition frequently generates overlapping contour segments.

A similar extraction algorithm is applied to select straight segments, which are indicated as squares in the upper right of figure 1.

*Contour Description.* The contours (partitioned and extracted) are then described by their structure (geometrically) and by their appearance. The parameterization returns the following parameters (see Rasche for details): orientation ($o$); length ($l$); arc ($a$), alternating (e.g. oscillating or wiggly; $x$);

curvature ($b$); edginess ($e$); symmetry ($s$); region ($r$); the mean and standard deviation of the contrast values along the contour ($c_m, c_s$); mean and standard deviation of the fuzziness values along the contour ($f_m, f_s$). The parameters are then combined to form the 12-dimensional vector $\mathbf{c}(o, l, a, x, b, e, s, r, c_m, c_s, f_m, f_s)$. In our previous study, we used a parameter transition ($t$), but which was dropped as only a small portion of contours can be assigned to this 'class'.

*2.1.2. Areas*

The symmetric-axis transform (SAT) - originally proposed by Blum (Blum 1973) - is generated in two steps. The 1st step consists of the propagation of contours to generate the distance map $DM$ (see figure 2b). Our implementation of the propagation process is particularly suited to generate this map for fragmented contour images (see [30]). The 2nd step consists of the convolution of this map with a band-pass filter followed by thresholding to select the symmetric axes of a structure. The sym-axes are then partitioned into their constituent, elementary segments at their points of intersection (figure 2c). The following parameters are then extracted (see section 3.2 [20] for details): orientation (o); angle ($\alpha$); elongation (e); mean symmetric distance ($s_m$); initial and end distance ($s_1$ and $s_2$); flexing distance ($s_{fx}$); flexing position ($p_{fx}$); curvature (b). The appearance parameters were the same ones as for the contour, but are taken from the enclosed area (and not along the contour segments), and contained a parameter for luminance range ($c_r, c_m, c_s, f_r, f_m, f_s$). In summary, we have the following 15-dimensional vector for a sym-axis segment, also called area: $\mathbf{a}(o, \alpha, e, s_m, s_1, s_2, s_{fx}, p_{fx}, b, c_r, c_m, c_s, f_r, f_m, f_s)$.

Due to the dimensionality problem (curse of dimensionality), we also tested a vector which contained only the appearance parameters in order to avoid geometric constraints. For instance, in natural scenes there are many areas containing category-typical textures, but whose geometry (of the area) do not possess a particular geometry. This texture descriptor $\mathbf{t}$ consisted of the dimension $z$ for the area size and the same 6 appearance parameters as

used in the above area vector:

$$\mathbf{t}(z, c_m, c_s, c_r, f_m, f_s, f_r). \tag{3}$$

The advantage of our implementation of the symmetric-axis transform is that it can very accurately determine the relation between two contours even though the contours are fragmented, which is expressed with the area vector. And it can outline a 'loose' area containing potentially category-characteristic texture, which is captured with the texture vector. There are however several short-comings with the SAT: 1) speckled noise or other contours prohibit the evolvement of a desired sym-axis. An implementation by Engel deals with such noise very well, but the resulting sym-axes do not correspond to the values of the distance map and that makes it difficult to determine the surrounding area [31]; 2) The SAT groups only neighboring contours and therefore groups locally only. Global grouping would consist for instance of grouping contours across other contours. Such a process we would assign to the synthesis process (next section), but has not been pursued yet in this study.

### *2.2. Synthesis*

The synthesis integrates the decomposition output (**c** and **a**) to more complex descriptors. The contour segments are integrated to a descriptor expressing local relations of two segments, e.g. L feature, parallel segments. The sym-axes segments are integrated to more complex descriptors expressing (fragmented) shapes.

#### *2.2.1. Local Pairings*

The creation of the following contour relations is very similar to Lowe's work on non-accidental groupings, but here only a fraction of these groupings is pursued [16]. As pointed out previously, the novelty of our approach is that a multi-dimensional space is formulated with those groupings. After the decomposition, most segments are either elongated, or just straight or curved segments, and can therefore be easily grouped. Two types of groupings are

pursued, or more specifically two types of pairings: adjacent segments, such as parallel or converging segments; and connected segments, such as L features or collinear segments. Those pairings are sought relatively local by using an acceptance threshold that is dependent on the shorter segment. In case of multiple potential pairings per contour, the pair with the shortest distance is selected, the remaining selections are discarded.

- *Adjacency*: Two segments $i$ and $j$ are considered as adjacent if their center points are proximal, specifically if the distance $d^{ij}_{cen}$ is below a chosen threshold. The threshold for proximity is a fraction of the shorter segment.

- *Connectedness*: The endpoints of two segments are considered as connected if their proximal distance $d^{ij}_{prox}$ is smaller than a fraction of the shorter segment.

The (local) pairing vector $\mathbf{p}$ is then formed with the following parameters: $\alpha$ is the angle between the two segments ($\alpha \in [0..\pi]$); $l_{mean}$ is the mean length of the two segments; $l_\delta$ is the normalized difference of the lengths ($|l_i - l_j|/l_{mean}$). And two distance values are added which in case of the connected pairs consists of the proximal and distal distance ($d_{prox}, d_{dist}$) and in case the adjacent pairs as the center distance ($d_{prox} = d_{dist} = d_{cen}$):

$$\mathbf{p}(\alpha, d_{prox}, d_{dist}, l_{mean}, l_\delta). \tag{4}$$

It would make sense to attempt to express also T junctions, but detecting potentially meaningful T pairings is computationally much more intensive as it required more distance measurements.

For many of those groupings, the decomposition has already generated an equivalent sym-axis segment - though the area vector is geometrically much more accurate. No effort was made to disolve this overlap, for instance to discard those double representations. The local pairings have the advantage not to suffer from the noise problem - as opposed to the generation of the sym-axes segments (see above criticism on SAT). The local-pairing vector has not been tested with appearance parameters yet.

*2.2.2. Regions*

For a shape, a sym-axes segment often represents a 'part'. Thus, the intersection of sym-axes segments often represents a joint of such parts. And by a description of those intersections and their segments, abstractions of more complex structure can be expressed. We call them now region descriptors, as the sym-axes represent region information better than the contours that generated them (for an accurate shape description it requires both, the contour and region information, see paragraph 'Description' in introduction of [20] for citations). Here, two types of regions descriptions are pursued. In one, the *junction* descriptor **i**, only the distance values of a circular surround are analyzed. The descriptor is rather local as it expresses only the immediate surround of the intersection. In the other type, the *skeleton* descriptor **k**, the geometry of the intersecting sym-axis segments is described, exploiting the segment's parameter values (of vector **a**). This descriptor can be global depending on the spatial extension of the segments and it is relatively complex.

*Junction.* A circle is placed at the intersection point with a radius corresponding to the distance value $s_c$ of the intersection point (grey stippled circle in figure 2c). The symmetric distance values along the circle's arc-length variable $k$, $DM(k)$, describe what we now call the *surround signature*. For a circle shape, the surround signature is flat because the values are taken along the contour where the distance values are 0. For a square shape, the signature shows 4 elevations which correspond to the crossing of the sym-axis segments of the L features. For a half-rectangle shape (see example in figure 2c and d), there are three elevations: the largest corresponds to the vertical segment and reflects the parallel segments; the two smaller ones correspond to the oblique segments reflecting the L features.

The surround signature allows to easily estimate the openness $u$ of the junction by dividing the signature's integral by its diameter: $u = \sum_k [DM(k)]/(2s_c)$ ($u = 0$ for a circle). To further specify the geometry we determine the logarithm $n$ of the number of elevations, the amplitude $\beta$ for each elevation and

the angle $\alpha$ between the locations of the amplitude maxima. For the list of amplitudes and angle values, we take the standard deviation, $\sigma_\beta$ and $\sigma_a$, respectively. The three largest amplitude and angle values were also selected as dimensions ($\beta_1, \beta_2, \beta_3$ and $\alpha_1, \alpha_2, \alpha_3$), thus forming an 11-dimensional junction vector **i**:

$$\mathbf{i}(s_c, u, n, \sigma_\beta, \sigma_a, \beta_1, \beta_2, \beta_3, \alpha_1, \alpha_2, \alpha_3). \tag{5}$$

For junctions consisting of more than 4 sym-axis segments the vector is less specific but such junctions are rare. This vector is tested without any appearance parameters as the appearance taken from just the intersection area is not as characteristic as the appearance as expressed by the texture vector for instance.

*Skeleton.* One could take the individual parameters of the (intersecting) sym-axes segments (as expressed by **a**), but that would be overly accurate and not serve well for the search of abstractions. We here concentrated in particular on the spatial extension of the parts and less so about the geometry of the individual parts. The dimension $s_c$ is the symmetric distance value at the point of intersection; dimension $n$ is the logarithm of the number of intersecting segments; dimension $z$ is the spatial (2D) area of the entire structure; dimensions $s_{min}$ and $s_{max}$ are the minimal and maximal distance value for the distal (outer) ends of the intersecting segments, which describe the openess of the shape; analogously, dimensions $l_{min}$ and $l_{max}$ are the minimal and maximal length of the segments; dimensions $l_{std}$ and $l_{mean}$ are the standard deviation and mean of the length values; dimensions $\alpha_{min}$ and $\alpha_{max}$ are the minimum and maximum angle.

In addition to those 11 geometric parameters, the same appearance parameters are added as for the area vector ($c_m, c_s, c_r, f_m, f_s, f_r$), taken from the region spanned by the intersecting segments. In summary, a 17-dimensional vector is used to describe skeletons:

$$\mathbf{k}(s_c, n, z, s_{min}, s_{max}, l_{min}, l_{max}, l_{mean}, l_{std}, \alpha_{min}, \alpha_{max}, c_m, c_s, c_r, f_m, f_s, f_r). \tag{6}$$

*2.3. Learning and Matching*

Two learning schemes were tested, that acquired 'category-specific' descriptors. In a search-learning scheme, the category-specific descriptors were found by using a similarity search. This scheme is the one we used in our previous study and because it took a relatively large sample number to successfully carry out this search (e.g. 10 images per category), we started to develop a novel learning scheme, the gradual-learning scheme, in which category representations from fewer images would be gradually developed.

Comparing two images occured by matching the descriptors $\mathbf{v}_j$ of one image against the descriptors $\mathbf{v}_i$ of the other image, resulting in a distance matrix $D_{ij}$. The shortest distance for each image is selected, e.g. $\mathbf{d}_i = \max_j D_{ij}$.

*Search-Learning Scheme.* A category representation was generated by searching for 'category-specific' descriptors. They were selected from a subsample of 10 images per category by a similarity search, whereby a descriptor $\mathbf{v}$ of one image was selected and a similarity search on all other images carried out ($\mathbf{v} \in \{\mathbf{c}, \mathbf{a}, \mathbf{t}, \mathbf{p}, \mathbf{i}, \mathbf{k}\}$). If the same category images appeared amongst the first few similar images, then the descriptor was kept as a category-specific descriptor and its degree of category-specificity, e.g. the proportion of same-category images within the first 100 images, is taken as its weight $w$.

*Gradual-Learning Scheme.* There are two stages in this learning process, an auto-correlation and a cross-correlation stage. The auto-correlation stage matches the descriptors of different sample images of the same category to determine the commonly (frequently) occuring descriptors - the equivalent to the category-specific descriptors (shown in figures 3 - 5). For three and more sample images, the common descriptors are collected from all pairs of auto-correlations and concatenated to a single list: identical contours amongst those pairs were discarded using a fixed threshold. The catgory representations grew slightly with the number of samples.

In the cross-correlation stage the common descriptors of one category are matched against the common descriptors of all other categories to determine

their category specificity ($w_i$) by counting the number of co-occurrences in the other categories ($w_i$ proportional to the lack of co-occurences).

*Matching.* In the testing (categorization) phase, the distance vector $\mathbf{d}_i$ between the image descriptors $\mathbf{v}_j$ and the category descriptors $\mathbf{v}_i$, was weighted using the category-specific descriptor weights $w$ forming the weight vector $\mathbf{w_i}$. Integration and normalization leads to the descriptor activity level, $a = \mathbf{d}_i \mathbf{w}_i' / \sum_i \mathbf{w}_i$, which in turn was integrated across scales and descriptors to form the category activity level, $A = \sum_{\sigma=1,2,3,5} \sum_{\{\mathbf{c}..\mathbf{k}\}} a$. A simple maximum search decided on the preferred category ($\max_i A_i$).

## 3. Implementation

*Preprocessing time.* The average processing times for a Caltech image at scale $\sigma = 1$ using an Intel 2GHz were: 390ms for the Canny algorithm; 392 ms for the appearance information (without region description); 1270ms for the extraction of contour segments from the image; 4000ms for the generation of the LG spaces and the derived spectrum and parameter description (increase of 1500ms in comparison to [20] due to the inclusion of the straightness label); 3000ms for contour propagation; 1000ms for local pairings; 956ms for sym-axis extraction, parameterization and formation of region vectors. Summarized, the entire preprocessing time (decomposition and synthesis) for an image is approximately 11.0 seconds (including inbetween saving of data files; increase of only 3.3 seconds to our previous study). For scale $\sigma = 5$, the average processing time is 4.9 seconds (3.4 seconds in previous study). The processing time for the learning procedures is given in the evaluation section.

The proximity threshold for adjacency was 0.6 of the shorter segment's length, the one for connectedness was 0.5 of that length.

## 4. Evaluation

The system was evaluated on the Caltech 101 and the Corel image set (see [20]. Four types of evaluations were carried out: three were carried out

as in our previous study (see 1 to 3); the fourth one used the novel learning approach, called the gradual-learning scheme, as explained in subsection 2.3.

1) Categorization using histograms: In the histogramming approach, a high-dimensional histogram was formed using all parameters, resulting now in a 620-dimensional vector in this study (62 parameters times 10 bins). The categorization performance was only 1-2 percent higher than in our previous approach (ca. 13 percent for the Caltech collection), showing that the use of many more parameters does not lead to much improvement. The knock-out showed again that all dimensions were relevant and that none was substantially more important than any other one (not presented).

2) Image search: In the similarity search task, images were ordered for a group of selected descriptors. The performance success was measured as the percentages of images of the same category appearing amongst the first 99 similar images. Carrying out this type of search with the novel descriptors raised the search performance by ca. 8 percent to ca. 24 percent for the Corel set and 29 percent for the Caltech set (as opposed to our previous study).

3) Categorization using search-learning scheme: this specific matching task was mentioned only marginally in our previous study as it did not achieve a substantially higher categorization performance as the histogramming approach. The selection of category-specific descriptors took several minutes for a Caltech category for all spatial scales ($\sigma$ =1, 2, 3 and 5) and a training sample number of 10 images per category; determining the category assignment took ca. 30 seconds. The performance for correct categorization was 19 percent for the Caltech image set.

4) Categorization using gradual-learning scheme (see subsection 2.3): The autocorrelation stage for two images (per category) occured within a duration of less than a second only per image, as it involved only matching the descriptors of a pair of images. The cross-correlation stage (determining $w$) took also less than a second as the number of category specific descriptors is smaller than the average number of image descriptors.

Figure 3 to 5 show the common descriptors for the auto-correlation of

two sample images for the texture, pairing and skeleton descriptor. The collection for category-specific contours and sym-axes looks very similar to the ones as when collected with the search-learning scheme (see figure 6 in [20]) and they are therefore not shown again. The descriptors here are shown at their original (spatial) location in the image, but no spatial information was used in the present evaluation.

The correct categorization performance was at ca. 14 percent for two sample images only, which we consider as very promising. However, for more sample images the performance increased only little and reached only 19 percent for five sample images. A similar performance level was obtained with the Corel collection.

Figure 6 shows the categorization performance for a variety of robustness tests. The upper left graph shows the performance when individual descriptors were omitted: for the straightness and pairing vector the percentage decreased most. That those are the most representative descriptors is also reflected in the performance level when individual descriptors were used only for matching (upper right graph). The intersection vector clearly had the least impact on overall categorization performance.

When individual spatial scales were knocked out (lower left graph), the performance dropped slightly only. But when used individually, they still showed a performance substantially above chance level (lower right graph).

## 5. Discussion

In our quest for abstractions for canonical patterns, we have created parameters and descriptors that are potentially useful for other recognition systems, that deal with structure, e.g. shape recognition systems [32] or object search systems [17, 18, 19]. It may well be that some of the parameters developed here are too accurate for representing canonical patterns but those could be necessary to discriminate between subtly different shapes as in [32].

The performance of our approach is still low as compared to other fast categorization systems [10, 11] because we have not tuned our system to

any specific image set yet and because a number of other descriptors are necessary (see list below). However, our methodology is more general as it attempts to describe any type of pattern and not just a specific set. The many parameters in our system leave room for weight tuning, if one intended to design a recognition system for a specific image set. Further weights could be added, for instance weights for the descriptor activity level, or for the spatial scale ($\sigma$), or combinations of descriptors.

Our methodology bears the possibility to acquire scene knowledge, e.g. knowledge about the spatial location of frequently occurring parts. Although there is a large effort to create such knowledge by labeling objects in scenes [33], our methodology offers to learn the typcial spatial location of objects in a completely unsupervised manner because many descriptors are very typical for a part of a scene or object, as demonstrated in figures 3 to 5. To move toward such 'scene' knowledge, each descriptor could be assigned a loose spatial location, whereby the horizontal locations is to be given a low weight due to the low bias of left/right ocurrences, and the vertical location (top/bottom) a higher weight.

. We do not regard our suggested descriptors as a complete list of descriptors but rather as an example of what type of multi-dimensional spaces may be created. Other descriptors that could be tested are the following:

- Groups of contours: an extension of the local pairing descriptor could be pursued which would group endpoints that are proximal (and not just pairs), thus describing any converging or intersecting contours.

- Groupings of sym-axes: in a similar way to the just mentioned groups of contours, sym-axes could be clustered, whose starting points are proximal, thus describing also vertex features (T junctions, X junctions,...).

Both types of groupings clearly overlap in the types of structures they represent, an overlap analogous to the one in local pairings and sym-axes segments. One solution maybe too simply leave this overlap, as it could provide the uniqueness and robustness to the ever-present noise and variability in structure (point 5 in subsection 1.1)

- Sequences of segments: Presently, a contour is described as either a curved segment, a straight segment, or an alternating contour (expressed by the dimensions $a$, $x$ and $b$). We realized that this is still a coarse description and possibly not sufficient to discriminate certain shapes, in which for instance inflexion segments are very characteristic. We had previously tried to describe such geometry by the transition parameter, which we dropped in this study because a contour rarely occurs as an isolated inflexion. Rather an inflexion segment is often part of an alternating contour. But with the extracted segments (straight and curved), one could build a 'polygon' descriptor that expresses a larger variety of contour geometries more distinctively. Such precision could be necessary to 'detect' an animal silhouette in a natural scene for instance, which together with texture features would classify the image as belonging to the category 'animal'.

- Groups of groups/structural relations: For some images, the above suggested grouping operations may already be of global nature and capture the canonical pattern sufficiently well. But for other images this may not suffice and further groupings amongst the above suggested groups are desirable to obtain more distinct category representations. Lastly, single structural relations between individual descriptors need to be tested, e.g. the distance and angular alignment between descriptors.

. The gradual-learning scheme is clearly more efficient than the search-learning scheme as it reached the same performance with half the sample images. The robustness test showed that the knock-out of some descriptors and scales did not lead to a large decrease in performance. It is tempting to assume that those descriptors could be omitted, but the individual performances show a relatively high level, demonstrating their representative power.

The performance of the intersection vector was lowest because it contains mere geometric information. In fact, it almost reaches the performance of the skeleton vector when the latter was tested with the geometric parameters only using the histogramming method (results not shown). This indicates that the intersection vector, whose geometrical information was solely taken

from the circular surround (figure 2c), represents already a useful abstraction of complex shapes, and that some form of classification can be done using the distance map only, without requiring the explicit analysis of sym-axes segments. The intersection vector would show its true merit in a shape classification task.

The overall preprocessing time (decomposition and synthesis) has increased only by one third, as the synthesis process is primarily the manipulation of vector lists, which can be performed faster than the decomposition processes.

The reconstruction process we propose is detailed at the level of contour geometry and is exhaustive in its attempt to find meaningful local and global relations. No specific attempt is made so far to segment an image, although further development of our system will show whether such processes are necessary for a perfect, fast category assignment.

## Acknowledgment

The study is supported by the Gaze-Based Communication Project (European Commission within the Information Society Technologies, contract no. IST-C-033816).

## References

[1] D. Marr, *Vision*. New York: W. H. Freeman, 1982.

[2] A. Witkin and J. Tenenbaum, "On the role of structure in vision," in *Human and machine vision*, J. Beck, B. Hope, and A. Rosenfeld, Eds. New York: Academic Press, 1983, pp. 481–543.

[3] J. Malik, S. Belongie, T. Leung, and J. Shi, "Contour and texture analysis for image segmentation," *International Journal Of Computer Vision*, vol. 43, no. 1, pp. 7–27, 2001.

[4] Z. Tu and S. Zhu, "Parsing images into regions, curves, and curve groups," *International Journal Of Computer Vision*, vol. 69, no. 2, pp. 223–249, 2006.

[5] J. Canny, "A computational approach to edge-detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[6] A. Bobick and R. Bolles, "The representation space paradigm of concurrent evolving object descriptions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 146–156, 1992.

[7] J. Elder, "Are edges incomplete?" *International Journal Of Computer Vision*, vol. 34, no. 2-3, pp. 97–122, 1999.

[8] Y. Wang and S.-C. Zhu, "Perceptual scale-space and its applications," *International Journal of Computer Vision*, vol. 80, pp. 143–165, 2008.

[9] Y. Keselman and S. Dickinson, "Generic model abstraction from examples," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 1141–1156, 2005.

[10] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.

[11] R. Fergus, P. Perona, and A. Zisserman, "Weakly supervised scale-invariant learning of models for visual recognition," *International Journal Of Computer Vision*, vol. 71, no. 3, pp. 273–303, Mar. 2007.

[12] F. Li, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, vol. 28, no. 4, pp. 594–611, 2006.

[13] A. Torralba, A. Oliva, M. Castelhano, and J. Henderson, "Contextual guidance of eye movements and attention in real-world scene: The role of global features on object search," *Psychological Review*, vol. 113, pp. 766–786, 2006.

[14] S. E. Palmer, E. Rosch, and P. Chase, "Canonical perspective and the perception of objects," in *Attention and performance IX*, J. Long and A. Baddeley, Eds. Hillsdale, NJ: Erlbaum, 1981, pp. 135–151.

[15] R. Brooks, "Symbolic reasoning among 3-d models and 2-d images," *Artificial Intelligence*, vol. 17, pp. 285–348, 1981.

[16] D. G. Lowe, *Perceptual organization and visual recognition.* Boston: Kluwer Academic Publishers, 1985.

[17] R. Nelson and A. Selinger, "A cubist approach to object recognition," in *Sixth International Conference on Computer Vision*, 1998.

[18] J. Shotton, A. Blake, and R. Cipolla, "Multi-scale categorical object recognition using contour fragments," *IEEE Transactions of Pattern Analysis and Machine Intelligence*, vol. 30(7), pp. 1270–1281, 2008.

[19] G. Heitz, G. Elidan, B. Packer, and D. Koller, "Shape-based object localization for descriptive classification," *International Journal of Computer Vision*, vol. 84, pp. 40–62, 2009.

[20] C. Rasche, "An approach to the parameterization of structure for fast categorization," *International Journal of Computer Vision*, vol. DOI 10.1007/s11263-009-0286-1, 2009.

[21] A. Bengtsson and J.-O. Eklundh, "Shape representation by multiscale contour approximation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, pp. 85–93, 1991.

[22] A. Guzman, "Analysis of curved line drawings using context and global information," in *Machine Intelligence 6*, M. Meltzer and D. Michie, Eds. Edinburgh, Scotland: Edinburgh University Press, 1971, ch. 20, pp. 325–375.

[23] M. Fischler and R. Elschlager, "The representation and matching of pictorial structures," *IEEE Transaction on Computer*, vol. 22, no. 1, pp. 67–92, 1973.

[24] P. Felzenszwalb and D. Huttenlocher, "Pictorial structures for object recognition," *International Journal Of Computer Vision*, vol. 61, no. 1, pp. 55–79, 2005.

[25] B. Schiele and J. Crowley, "Recognition without correspondence using multidimensional receptive field histograms," *INTERNATIONAL JOURNAL OF COMPUTER VISION*, vol. 36, no. 1, pp. 31–50, 2000.

[26] H. Asada and M. Brady, "The curvature primal sketch," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, pp. 2–14, 1986.

[27] F. Mokhtarian and A. Mackworth, "Scale-based description and recognition of planar curves and two-dimensional shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, pp. 34–43, 1986.

[28] ——, "A theory of multiscale, curvature-based shape representation for planar curves," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 789–805, 1992.

[29] A. Torralba, R. Fergus, and W. T. Freeman, "80 million tiny images: a large dataset for non-parametric object and scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 2008*, vol. 30(11), pp. 1958–1970, 2008.

[30] C. Rasche, "Neuromorphic excitable maps for visual processing," *IEEE Transactions on Neural Networks*, vol. 18, no. 2, pp. 520–529, 2007.

[31] D. Engel and C. Curio, "Scale-invariant medial features based on gradient vector flow fields." *ICPR 2008, December 8-11, 2008, Tampa, USA*, 2008.

[32] M. R. Daliri and V. Torre, "Classification of silhouettes using contour fragments," *Computer Vision and Image Understanding*, vol. 113, pp. 1017–1025, 2009.

[33] B. Russell and A. Torralba, "Labelme: a database and web-based tool for image annotation," *INTERNATIONAL JOURNAL OF COMPUTER VISION*, vol. 77, pp. 157–173, 2008.

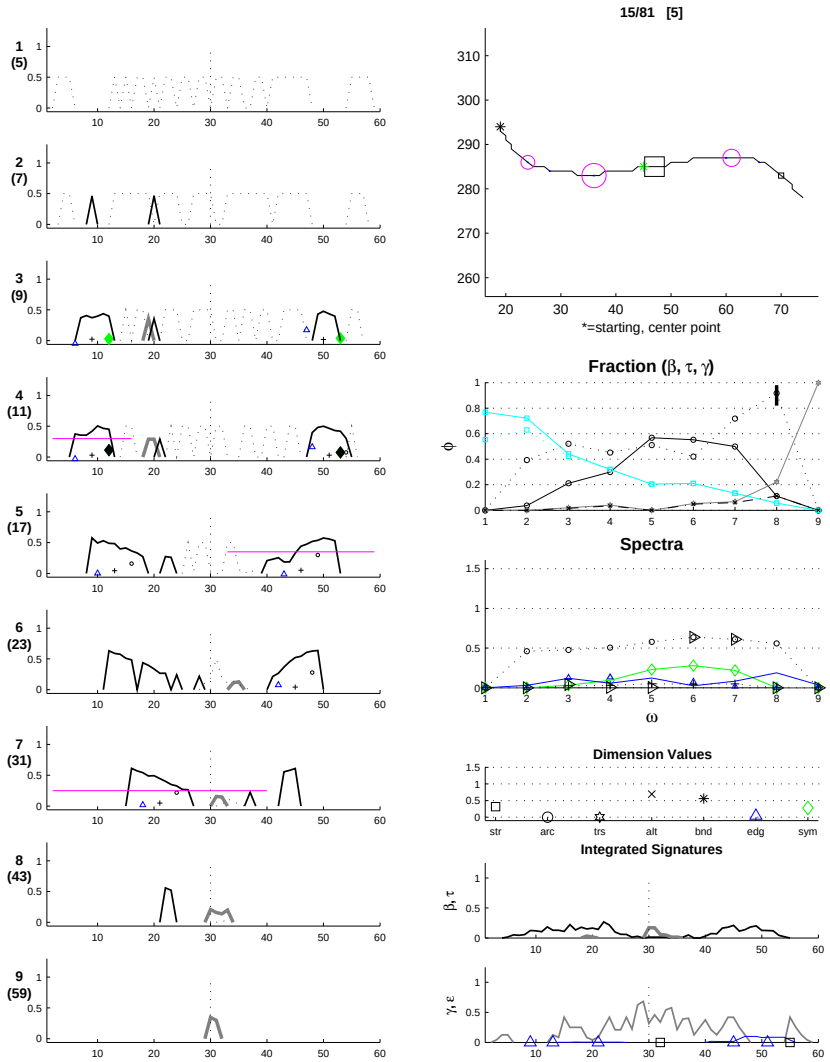


Figure 1: Local/global (LG) space. **Top right**: sample contour with starting and center points marked as asterisk; squares and circles denote straight and bow segments. **Left column**: LG space: signatures $\beta(v)$ (black), $\tau(v)$ (grey) and $\gamma(v)$ (stippled) for 9 different window sizes [x-axis= arc length variable $v$]. Function block characteristics (determined for large ones only): blue marker= $\epsilon^{\sqcap}$; green diamond=$v^{\sqcap}$; plus sign=$\zeta^{\sqcap}$. Extracted segments shown with horizontal line at a value of ca. 0.3 (window no. 4, 5, 7). **Fraction**: fraction functions $\phi$ for bowness, inflexion and straighness. **Spectra**: Green diamond: maximum of symmetry value; black circle: maximum $\beta$ amplitude; plus sign: maximum of $\zeta$. **Dimension Values**: straightness (not used), arc, transition (not used), alternation, bnd=curvature, edginess, symmetry. **Integrated Signatures**: bowness (black), transition (gray) (top graph); edginess (blue) and straightness (gray) (bottom graph).

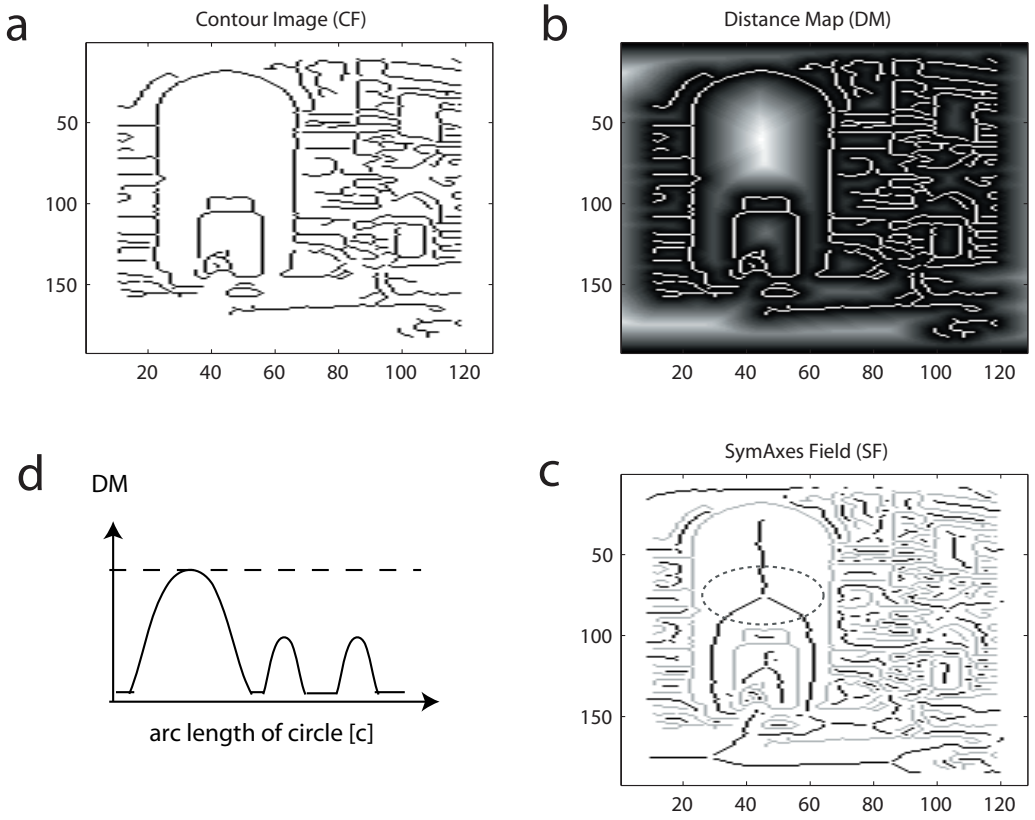


Figure 2: The surround signature. The signature is obtained from a circular 'mask' centered at the point of intersecting sym-axis segments. **a**. Contour image. **b**. Distance map (contous in white). **c**. Symmetric-axis field in black (contours in gray). The map is already partitioned into sym-axis segments at points of intersections (sym-points at intersections omitted in this graph). Example of a circular mask (stippled, center point at x=43 and y=85). **d**. Schematic of the signature of the example in c: the large bump corresponds to the vertical sym-axis segment, the two smaller bumps to the 'oblique' segments. The dashed line corresponds to the symmetric distance at the point of intersection.

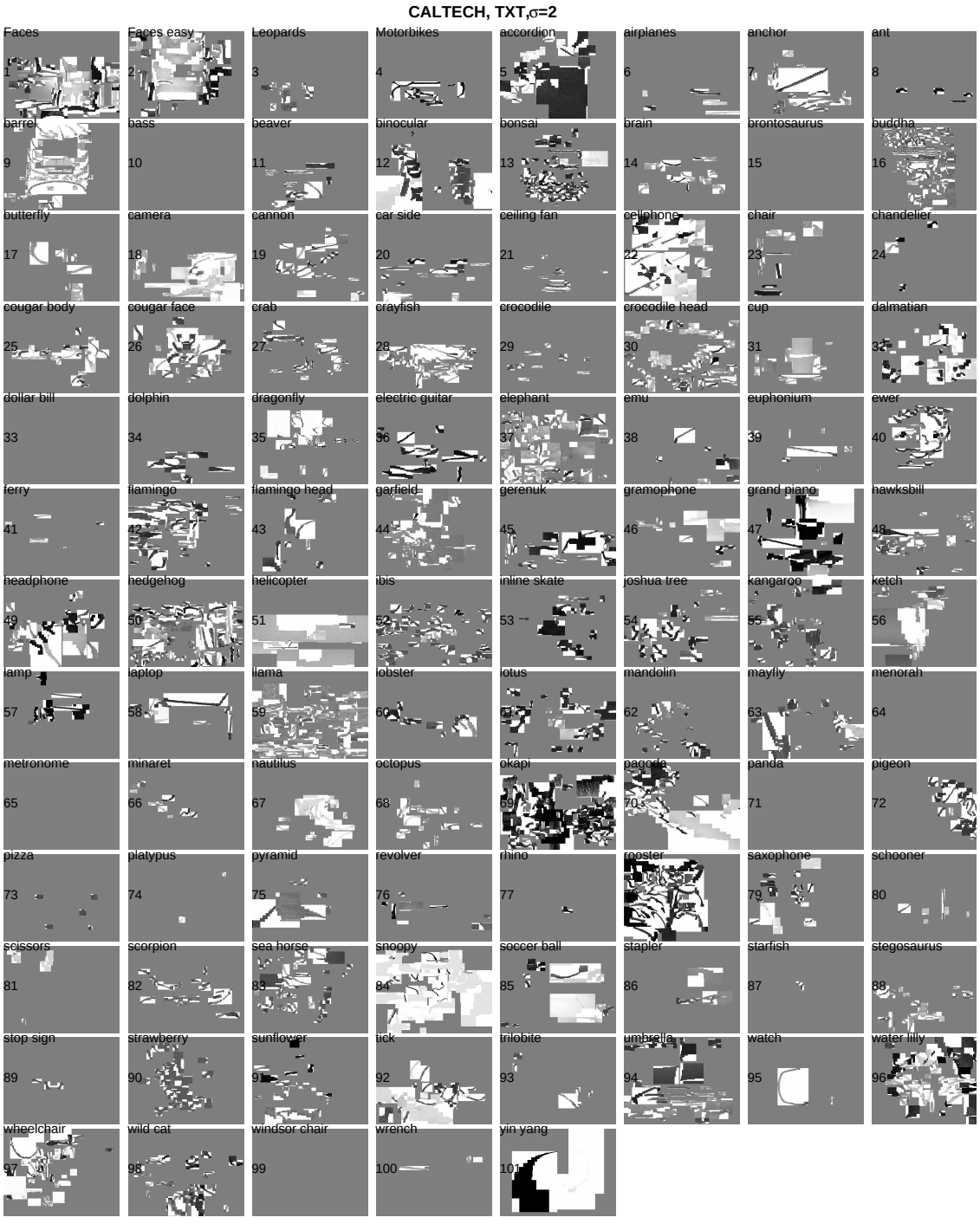


Figure 3: Category-specific texture descriptors from spatial scale $\sigma = 2$ for all 101 categories of the Caltech collection as determined from 2 images per category. The texture describes simplest appearance statistics taken from an area outlined by a symmetric-axis segment.

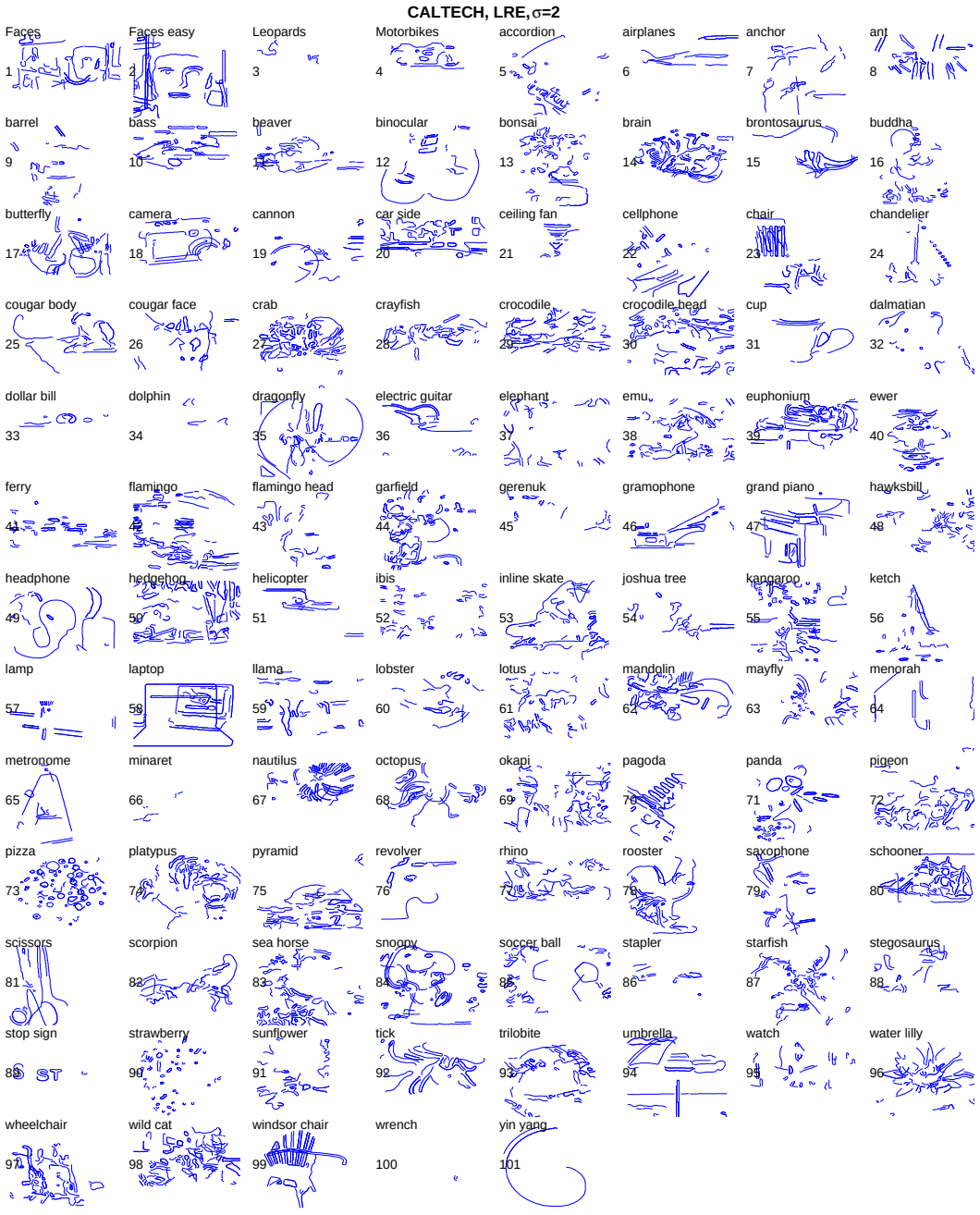


Figure 4: Category-specific descriptors for local contour pairings from spatial scale $\sigma = 2$ for all 101 categories of the Caltech collection as determined from 2 images per category.

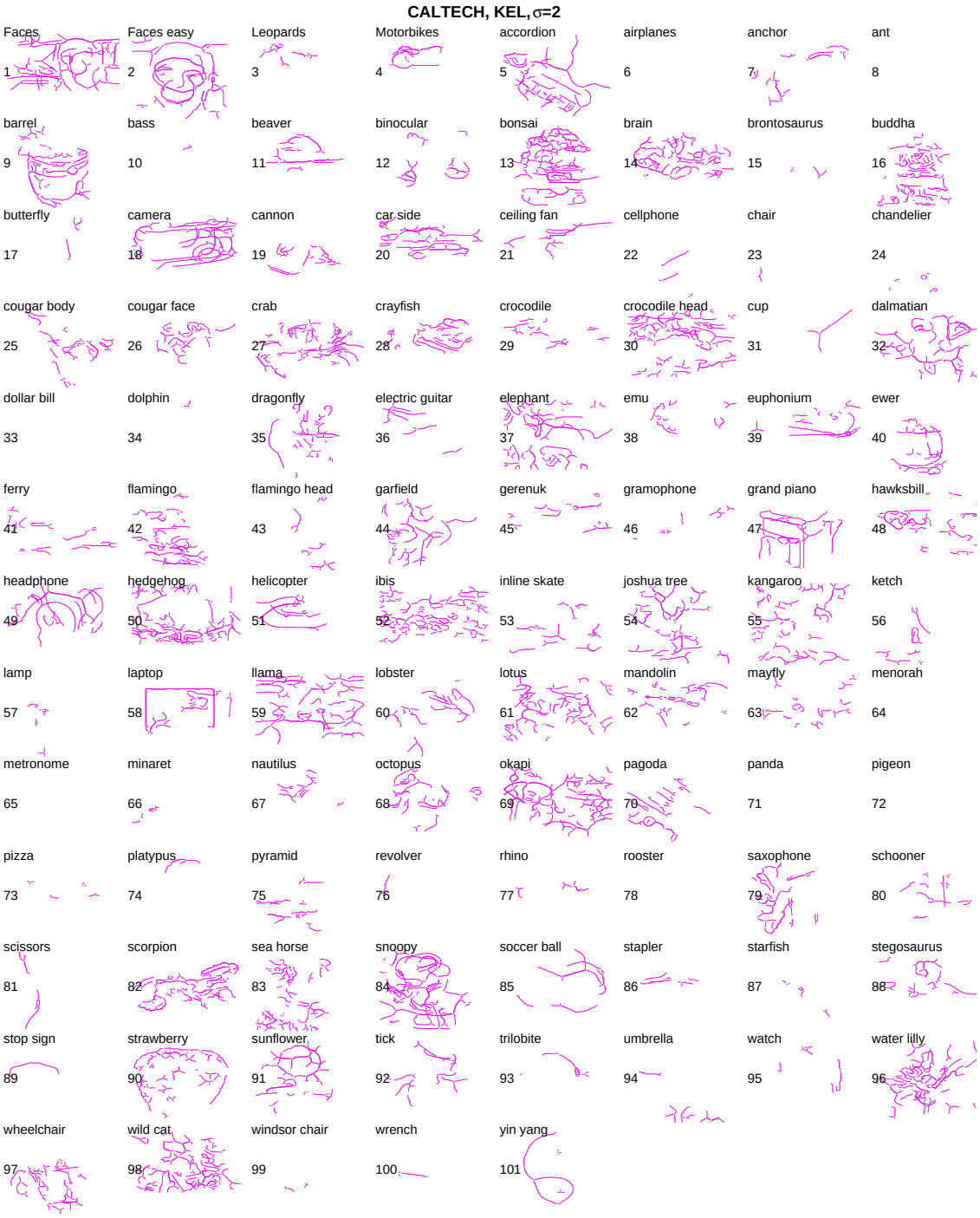


Figure 5: Category-specific skeleton descriptors (intersecting sym-axes segments, see for instance figure 2c) for spatial scale $\sigma = 2$ for all 101 categories of the Caltech collection.

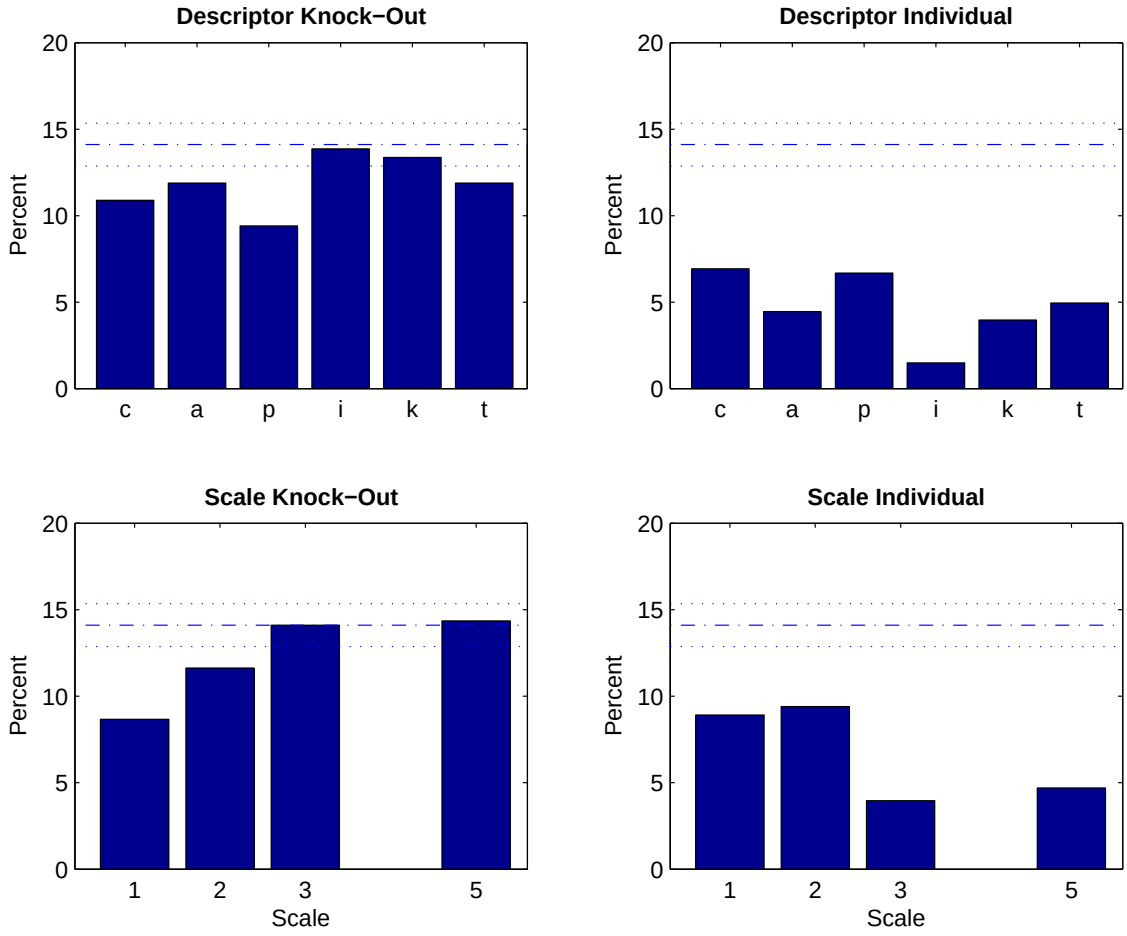


Figure 6: Categorization performance for learning with three sample images. Descriptor Knock-Out: performance when one descriptor is omitted. Descriptor Individual: performance for a single descriptor. Scale Knock-Out: performance when one spatial scale is omitted. Scale Individual: performance for a single spatial scale. The dashed line is the average categorization performance (total); stippled lines correspond to one standard deviation.