

# THE POTENTIAL OF CONTOUR GROUPING FOR IMAGE CLASSIFICATION

First Author Name, Second Author Name

*Institute of Problem Solving, XYZ University, My Street, MyTown, MyCountry*

*f\_author@ips.xyz.edu, s\_author@ips.xyz.edu*

Keywords: image classification, contour description, contour grouping

Abstract: An image classification system is introduced, that is predominantly based on a description of contours and their relations. A contour is described by geometric parameters characterizing its global aspects (arc or alternating) and its local aspects (degree of curvature, edginess, symmetry). To express the relation between contours, we use a multi-dimensional vector, whose parameters describe distances between contour points and the contours' local aspects. This allows comparing for instance L features or parallel contours with a simple distance measure. The approach has been evaluated on two image collections (Caltech 101 and Corel) and shows a reasonable categorization performance, yet its future lies in exploiting the preprocessing to understand 'parts' of the image.

## 1 INTRODUCTION

Recent approaches to image classification have used a variety of methods for their success. For instance Oliva and Torralba use the Fourier transform to preprocess gray-scale images of outdoor scenes (urban and natural), whose spectra are then classified (Oliva and Torralba, 2001); the group by Perona uses the principal component analysis to classify rigid objects with clear silhouettes (the Caltech-101 collection (Fergus et al., 2007)); others achieve comparable performances using histograms of selected features (Perronnin et al., 2006), systematic image histogramming (Lazebnik et al., 2006) and inspiration by Gestalt laws (Bileschi and Wolf, 2007). These approaches are good at discriminating image categories, but once the image is classified, their preprocessing output does not allow to analyze the structure, for instance to understand parts of the image or to determine the orientation of the recognized object. To carry out such a structural analysis it required a novel processing of the image. For instance, in case of the spatial envelope system by Oliva and Torralba, a preprocessing based on local orientations was developed, that allows for a visual search (Torralba et al., 2006), an effort which appears to be a move toward a struc-

tural description. Clearly, a structural description is still the most promising approach to a complete scene understanding system.

The idea of structural description is typically associated with an exact reconstruction of the image, starting for instance with image segmentation. This direction is perceived as little promising for the task of image classification given the wave of above mentioned attempts to 'directly' classify (see also explicit arguments by Oliva and Torralba in (Oliva and Torralba, 2001)). Furthermore, contours often appear fragmented and seemingly do not allow for a straightforward assignment to their 'parts'. Still, contour and structural-description approaches show their promise in object-search systems, for instance as templates of a 'Cubist' representation (Nelson and Selinger, 1998) (see also (Shotton et al., 2008; Opelt and Prinz, 2006; Zheng et al., 2007)); others develop learning algorithms for edge detection for specific image sets (Dollr et al., 2006); or try to use contour information to detect junctions in natural images (Maire et al., 2008). In this study, a structural description is pursued that describes contour geometry and their relations very accurately; elaborate image segmentation is avoided with the consequence of producing frequent accidental detections, which however are made

negligible by a matching process using redundant category representations. But because structure is described exhaustively, it potentially allows a detailed image analysis without the need of a novel preprocessing.

One way to relate contours is to use the symmetric-axis transform (Blum, 1973). Though conceptually elegant, it suffers from susceptibility to speckled noise which leads to distortions of the sym-axes, and it generates local relations only, meaning the sym-axes are formed only between immediately neighboring contours. However, what it also requires are global relations (or groupings) between contours, whereby irrelevant contours may lie in-between (e.g. speckled noise). The study by Bileschi and Wolf is a step toward that direction (Bileschi and Wolf, 2007): it relies on finding grouping principles from pixel correlations, that however are time intensive (ca. 80 secs/image). Instead, a grouping by contours would be less intensive and potentially more powerful, yet has been hardly pursued. That is the novelty of this study. To pursue such a contour-based approach, it requires a method which can reliably identify contours. We thereby use the method described in (Rasche, 2009), that is summarized in subsection 2.1. Subsection 2.2 explains the grouping procedure tested in this study.

## 2 MODEL

### 2.1 Contour Description, Partitioning and Extraction

The contour description is derived from distance distributions that in turn are obtained from systematic measurements along the contour. For an arbitrary contour a so-called local/global (LG) space is created, which is a description analogous to the scale space (fine/coarse space) but does not involve low-pass filtering (Rasche, 2009). The contour's global geometry is classified into either arc ( $a$ ) or alternating ( $w$ ), whereby the values are scalar and express the strength of these aspects. The contour's local aspects are described by the curvature parameter ( $b$ ), that expresses the circularity and amplitude of the arc and alternating contour respectively; the edginess parameters, that expresses the sharpness of a curve (L feature or bow); the symmetry parameter, that expresses the 'evenness' of the contour.

Contours are partitioned as follows: if a contour contains an 'end' - a turn of 180 degrees - it is partitioned at its point of highest curvature. After ap-

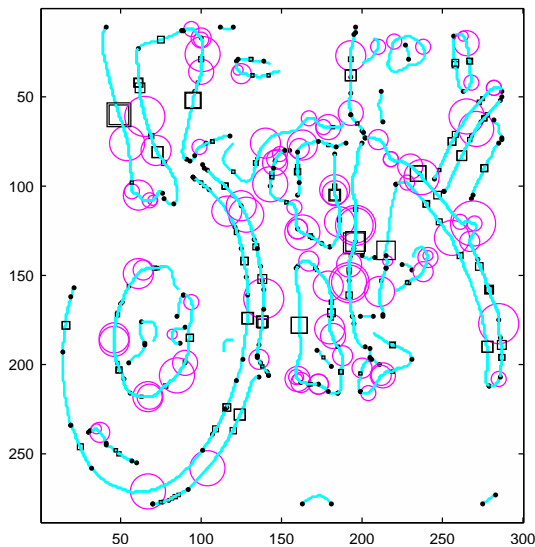
plication of this rule, any contour appears either as elongated in a coarse sense and can thus be classified as either alternating or curved ( $w$  or  $a$ ). An exception to this rule are smooth arcs, whose arc length is larger than 180 degrees; they are extracted separately. Exemplifying these two partitioning steps on the  $\Omega$  shape: the shape is havened and its circular part is extracted.

An alternating contour may span several objects (or parts) and that can be very characteristic to a category (as for instance the vertical wiggly contour for a person). Yet, its individual curved and straight segments can form potentially useful groupings with other contours of the structure. Thus, further partitioning for the purpose of grouping meant also losing potential category specificity. In this study, such alternating contours are not further partitioned but any straight or reasonably smooth, curved segment of sufficient arc length is extracted from it. Such elementary segments can be identified using the LG space. For a wiggly, natural contour such segments hardly exist, but many object silhouettes contain multiple such segments. Thus, the decomposition process does not strictly partition the contours into separate segments, but will create partially overlapping segments to some extent. Taking the  $\Omega$  shape as the example again, it is partitioned into 5 segments: one smooth arc segment; two L features representing the corners; and two straight segments (if of sufficient arc length).

The left graph in Figure 1 shows an example output of this decomposition. The long smooth arc outlining the wheel shows multiple segment extractions (straight and curved) because a) the segment is an ellipse, and b) due to the aliasing problem and the associated difficulty of discriminating between a smooth arc and a circularly aligned (open) polygon, that is discriminating between circle and hexagon for instance. Filtering techniques could resolve this latter issue (Lowe, 1989), but would introduce additional computation time, which we think can be avoided as we merely intend to find the semantic content of the image and not a precise reconstruction. In addition to those contours (denoted as  $c$ ), we use the symmetric-axis descriptors  $a$  as in (Rasche, 2009).

### 2.2 Grouping

Relating all contours with each other is excessive and leads to an unspecific structural description. Thus, there need to be some constraints that reduce the number of all possible relations to a smaller set of meaningful groupings. Such constraints have already been described by Lowe for the purpose of determining the orientation of objects in 3D space (Lowe, 1985). For



Smooth Arcs and Straight Segments (img=8904)

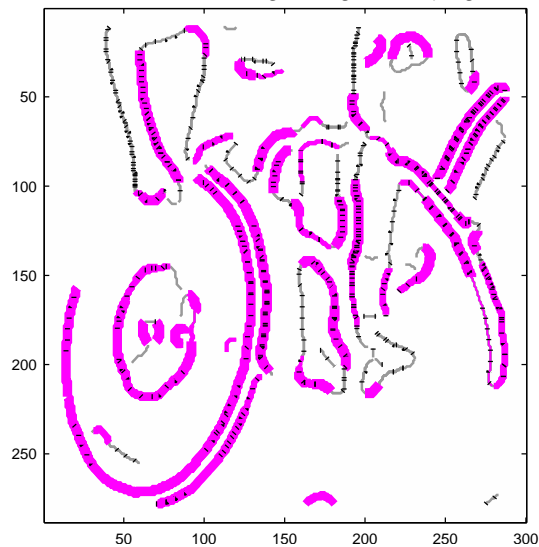


Figure 1: Decomposition output for an image (wheel chair from Caltech-101 collection). **Left:** Contour endpoints are marked as small black circles; squares and circles denote straight and curved segments respectively (their size reflects segment length - not curvature). Overlapping circles result from the global-to-local identification of elementary segments. **Right:** Smooth arcs (thick gray [magenta]) and straight segments (thick stippled) after application of a simple smoothness criterion to individual contours.

instance, closely spaced contour endpoints or parallel lines are 'salient' groupings which point to certain object poses. In case of a description for an arbitrary structure, the issue of grouping is more complex as essentially any spatial arrangement of two contours can be very category specific. Thus, the aim is therefore to find criteria that eliminate irrelevant pairings and keep the potentially category-characteristic pairs. In this study, we tested the following criteria:

1) Smoothness: only reasonably smooth contours were allowed for pairs, by choosing segments ( $\mathbf{c}$ ) with a low edginess value (see Figure 1 right graph). See also (Felzenszwalb and McAllester, 2006) for a method of finding salient curves.

2) Nearest Choice: For a given contour, only the two most proximal contours are admitted as pairs. Proximity is determined for the two endpoints and the center point of a contour.

Those two criteria reduce the number of pairings substantially, but they may already eliminate some category-characteristic pairs, in particular the nearest choice criterion as there could exist distinct pairs on a global scale. We therefore used one pairing that is salient independent of the intersegment distance of the two segments:

3) Closure: A contour pair that appeared as round or as encapsulating an area, e.g. two curved segments lying on opposite sides of a circle.

A contour-pair vector is created consisting of the following parameters: the distance between the proximal endpoints ( $d_c$ ); the distance between the center points ( $d_c$ ); the distance between the distal endpoints ( $d_o$ ), average contour length ( $l$ ), the asymmetry of contour lengths ( $y$ ); the curvature values for the two contours ( $b_1$  and  $b_2$ ; obtained from the contour description):

$$\mathbf{p}(o, d_c, d_m, d_o, l, y, b_1, b_2). \quad (1)$$

We also tested a texture descriptor  $\mathbf{t}$  consisting of the appearance dimensions of the sym-axis descriptor only. And we also tested a 'cluster' descriptor  $\mathbf{r}$ , which expresses the geometry of a contour cluster in a statistical sense. Those descriptors are not explained further for reason of brevity.

### 3 IMPLEMENTATION & EVALUATION

The closure criterion was implemented by choosing pairs whose distance between the center points was larger than the distances between the end points by a factor of 1.1; this is a simple but somewhat loose rule, selecting also a small number of 'distorted' pairings (e.g. segments not facing each other symmetrically).

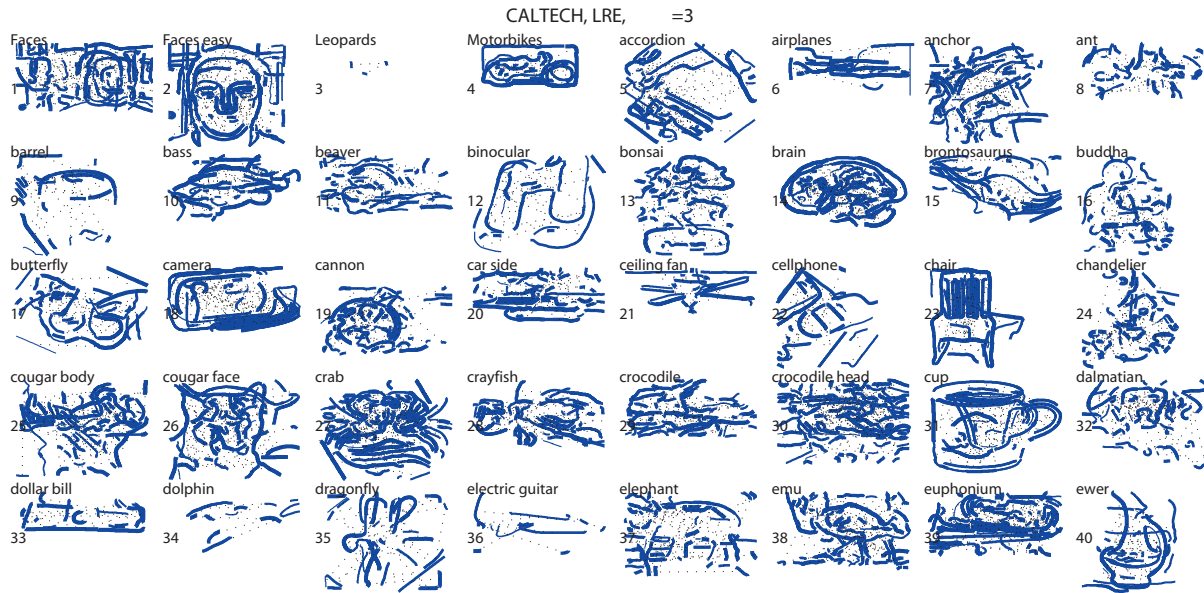


Figure 2: Category-specific contour pairs for the first 40 categories of the Caltech-101 collection (using two training images). The thin dotted line connects a pair; contour thickness corresponds to the descriptor weight. A pair is obtained by comparing the individual vectors of two images (of the same category).

The preprocessing was carried out for four (spatial) scales ( $\sigma=1,2,3,5$ ). The average operation duration for a Caltech image at scale  $\sigma = 1$  using Matlab on an Intel 2GHz was: 1300ms for creating the contour-pair vectors; 940ms for the cluster vectors; another ca. 8 seconds were used for contour extraction, description and sym-axes generation (total ca. 10 secs). For scale  $\sigma = 5$ , the average image-preprocessing duration was 4.8 seconds; for all 4 scales it was ca. 30 seconds.

The model was evaluated on the Caltech 101 collection (Fergus et al., 2007) and the COREL collection (see e.g. (Rasche, 2009) for its use). In a learning phase, category-specific descriptors (the 'Cubist' representation) were determined by finding similar descriptors amongst two images of the same category (see Figure 2 for contour pairs). Descriptor weights were set by 'cross-correlating' the Cubist representations and determining how often (rare) they occur in any other categories.

In a testing (categorization) phase, the descriptors of each Cubist representation are matched against the descriptors of a test image. More specifically, the descriptors  $\mathbf{v}_j$  of a test image were matched against the category-specific descriptors  $\mathbf{v}_i$  of a category, resulting in a distance matrix  $D_{ij}$ . The shortest distance for each category-specific descriptor was selected  $\mathbf{d}_i = \max_j D_{ij}$  and multiplied with the descriptor weights. A simple integration across descriptors

and scales, followed by a maximum search decided on the preferred category.

For two training images, the performance was ca. 19 percent; the average ranking value was ca. 16, that is the rank number at which the correct category appears (1=correct categorization; 51=random ranking). The left graph in Figure 3 shows the average ranking for all categories: about half of the categories were ranked among the top 18; the last 10 categories seemed randomly ranked (value around 51). The average descriptor weight was highest for the area vector, likely because the descriptor contained the largest number of dimensions (12). Given the small number of dimensions for the contour pair vector, its average weight was relatively high.

The performance for individual descriptors and scales, as well as knock-out (leave-one-out) simulations is depicted in Figure 4. The contour-pair descriptor had the largest impact on performance, which is evidenced by the large individual performance (ca. 13 percent, see 'Descriptor Individual') and by the significant performance decrease for a knock-out simulation (ca. 10 percent, see 'Descriptor KnockOut'). Lower (spatial) scales were more effective than higher spatial scales, e.g. 14 and 15 percent for scales 1 and 2.

For the COREL collection, the categorization performance was slightly lower (17 percent, two training images) and the performance pattern for the various

tests looked similar.

A learning process for 5 images was also tested for the Caltech collection only. Firstly, category-specific descriptors for each pair of images are determined, followed by pooling the ones that were different amongst image pairs. The number of features increased by ca. 40 percent, but the performance increased only by 25 percent (Figure 4b), a rather marginal increase. However the performance of individual descriptors increased by several folds except for the pairing vector.

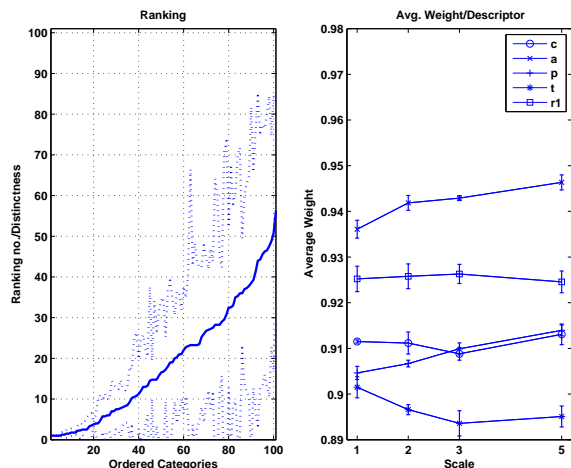


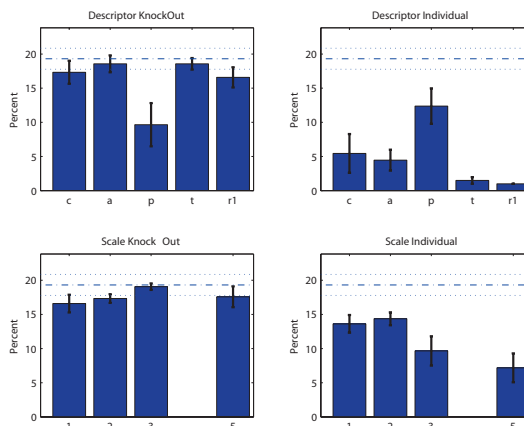
Figure 3: Performance analysis for 2 training images. **Left:** Average ranking with corresponding standard deviation (dotted). **Right:** Average weight per descriptor and scale. Error bars: standard deviation of crossfolds.

Figure 5 shows the 7 most similar categories for the 10 best- and worst-ranking categories. Even for the worst-ranking categories the similar categories can show structural similarities; sometimes, the similar categories correspond to the same super-ordinate category, e.g. animal categories would select other animals as similar categories.

## 4 DISCUSSION

The overall categorization performance (19 percent for 2 training images) is not quite comparable yet to other categorization attempts (up to 50 percent for 2 training images, see citations in introduction), but the study demonstrates the power of expressing contours as vectors and relating them by simple vector calculation. Furthermore and more importantly, the present approach bears the possibility to interpret the preprocessing output if a more detailed understanding of the image is desired, e.g. the parameters describe very ac-

### a 2 training images



### b 5 training images

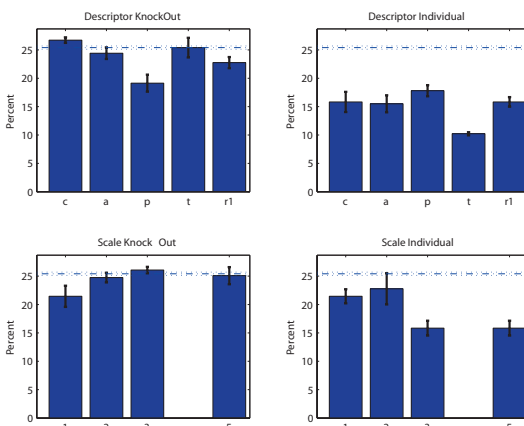


Figure 4: Performance (correct categorization) for 2 and 5 training images (a and b) for descriptor knock-out (upper left), individual descriptors (upper right), spatial scale knock-out (lower left) and individual scale (lower right). Note the performance increase for individual descriptors for 5 training images (compare upper right graph of a and b). Stippled horizontal line: total performance; dotted lines and error bars: standard deviation of crossfolds.

curately the geometry and spatial location of contours and areas. This is a specificity that none of the other image classification systems provides.

One potentially simple way to increase the performance is to combine our approach with an approach that is based on appearance, e.g. (Perronnin et al., 2006; Lowe, 2004). However, to implement a complete image understanding system, it requires the structural thoroughness as pursued here. There are many other sites, where the system can be improved but we expect the largest performance increase from the following improvements: a) refined grouping, for instance a grouping by geometrical similarity; b) fur-



Figure 5: Best and worst performing categories and their most similar categories, top ten and bottom ten rows respectively (Caltech 101). The first image in each row, represents a randomly selected image of that category; the remaining 7 images represent the most similar categories with increasing distance.

ther distance measurements and parameterization, for instance clusters of intersections; c) a probabilistic formulation of the presence of descriptors for a category representation; d) a better learning process.

## REFERENCES

Bileschi, S. and Wolf, L. (2007). Image representations beyond histograms of gradients: The role of gestalt descriptors. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 18-23, 2007, Minneapolis, USA*, pages 1–8.

Blum, H. (1973). Biological shape and visual science .1. *Journal Of Theoretical Biology*, 38(2):205–287.

Dollr, P., Tu, Z., and Belongie, S. (2006). Supervised learn-

ing of edges and object boundaries. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2006*, 2:1964–1971.

Felzenszwalb, P. and McAllester, D. (2006). A min-cover approach for finding salient curves. *IEEE Conference on Computer Vision and Pattern Recognition 2006*, pages 185–185.

Fergus, R., Perona, P., and Zisserman, A. (2007). Weakly supervised scale-invariant learning of models for visual recognition. *International Journal Of Computer Vision*, 71(3):273–303.

Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Lowe, D. (1989). Organization of smooth image curves at multiple scales. *International Journal of Computer Vision*, 3:119–130.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *INTERNATIONAL JOURNAL OF COMPUTER VISION*, 60(2):91–110.

Lowe, D. G. (1985). *Perceptual organization and visual recognition*. Kluwer Academic Publishers, Boston.

Maire, M., Arbelez, P., Fowlkes, C., and Malik, J. (2008). Using contours to detect and localize junctions in natural images. *IEEE Conference on Computer Vision and Pattern Recognition 2008*, pages 1–8.

Nelson, R. and Selinger, A. (1998). A cubist approach to object recognition. In *Sixth International Conference on Computer Vision*.

Oliva, A. and Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. J. Comput. Vis.*, 42(3):145–175.

Opelt, A. and Prinz, A. (2006). Fusing shape and appearance information for object category detection. *British Machine Vision Conference (BMVC) 2006*.

Perronnin, F., Dance, C., Csurka, G., and Bressan, M. (2006). Adapted vocabularies for generic visual categorization. *European Conference on Computer Vision, Graz, Austria 2006*, 4:464–475.

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

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

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

Zheng, S., Tu, Z., and Yuille, A. L. (2007). Detecting object boundaries using low-, mid-, and high-level information. *IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, USA 17-22 June 2007*, pages 1–8.