Shape Detection with Nearest Neighbour Contour Fragments

Kasim Terzić
w3.ualg.pt/~kterzic
Hussein Adnan Mohammed
a48025@ualg.pt
J.M.H. du Buf
w3.ualg.pt/~dubuf

Vision Lab (LARSyS)
University of the Algarve
Faculdade de Ciências e Tecnologia
Gambelas, Faro, Portugal

Abstract

We present a novel method for shape detection in natural scenes based on incomplete contour fragments and nearest neighbour search. In contrast to popular methods which employ sliding windows, chamfer matching and SVMs, we characterise each contour fragment by a local descriptor and perform a fast nearest-neighbour search to find similar fragments in the training set. Based on this idea, we show how to learn robust object models from training images, to generate reliable object hypotheses, and to verify them. Despite its extreme simplicity and speed, our method produces competitive detection results on the challenging ETHZ dataset.

1 Introduction

Shape is probably the single most important feature for object detection and much research has gone into developing deformable shape models. However, contours extracted by bottom-up edge detectors are notoriously unreliable, especially in natural images.

Traditional shape modelling methods such as Shape Context perform really well when complete and closed contours are available [1], but edges obtained from natural images of complex scenes are rarely complete or clean; one never knows where an incomplete edge fragment will begin or end, complicating the matching process. Even state of the art bottom-up edge extraction methods [12] will rarely produce complete, clutter-free contours in natural images. Instead, complete shapes tend to get split into many incomplete *contour fragments* which are mixed with unrelated "clutter" fragments produced by other objects, background, or texture. Additionally, any part of the object contour may be missing, complicating any gestalt-based contour merging. This means that model-based, top-down matching becomes important.

A number of algorithms have been proposed to solve the general shape detection problem in natural images. Typically, they consist of four stages: feature extraction, model creation, hypothesis generation, and hypothesis verification (or scoring). Some stages may be missing. For example, a system may apply a sliding window instead of generating hypotheses, it may use a hand-drawn model, or hypothesis generation may already provide a score as is

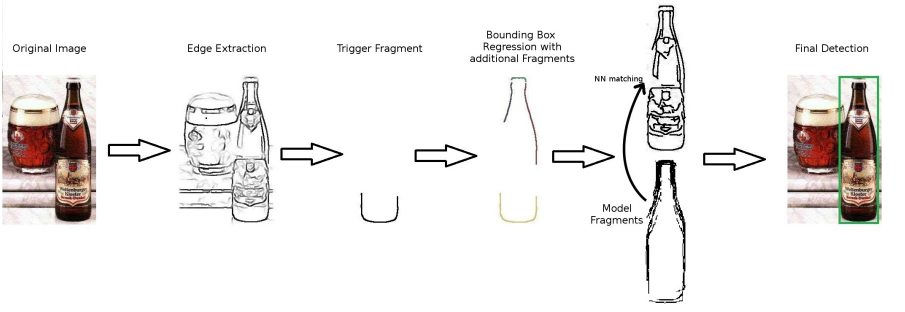


Figure 1: Overview of our method. Left to right: Original image, edges extracted by the Berkeley detector, a trigger fragment which generates a "bottle" hypothesis, several other bottle-like segments from that hypothesis used to refine the bounding box, matching a model (below) to contour fragments within the bounding box to obtain a score for the hypothesis and finally, a detected object.

the case in many voting methods. While excellent progress has been achieved in terms of experimental evaluation, much of the research has gone into developing more powerful descriptors [17, 30] or speeding up brute-force parts of the algorithms such as the chamfer matching step [9].

In this paper, we propose an orthogonal approach and present a very simple, yet powerful method for model creation, hypothesis generation, and hypothesis verification, which is competitive with much more complex methods. We use simple contour following combined with a standard shape context descriptor, and compare contour fragments using an efficient nearest neighbour algorithm [15]. We show that a simple and consistent nearest-neighbour approach can learn models, generate, refine and verify hypotheses, and achieve results comparable to the state of the art on the entire ETHZ dataset in about one hour in total. The main contributions of this paper are: i) a novel shape model based on intuitive Bayesian parameters, ii) a simple and consistent model based on nearest neighbours, and iii) fast runtime with minimal learning.

2 Related Work

Shape detection in natural images is usually based on incomplete contour fragments automatically extracted by an edge detector [3, 12]. In this line of research, it is widely accepted that a single incomplete contour fragment is not sufficient for reliable detection due to occlusions, broken contours and clutter. Hence, some approaches attempt to explicitly model the relationships between several contour fragments. Ferrari *et al*. [6] introduced the concept of Pairs of Adjacent Segments, and Yarlagadda and Ommer [30] learned joint placements of several relevant contour fragments . Other researchers tried to merge contour fragments to obtain longer and more salient contours [10]. Often, a complete object model is mapped into the image using fast directional chamfer matching and the total distance is used as a scoring criterion [9].

In addition to the contour-based approach, local image descriptors known from object recognition have also been successfully applied, such as HOG variants [4, 14] and Geometric Blur [17]. This group of methods does not explicitly model contours, but relies on the

implicit shape representation provided by the descriptors. A powerful classifier such as an SVM then learns the important shape characteristics. Another popular approach is to use regions [8, 20] and super-pixels [27]. Region segmentation tends to split the object into several larger regions, as opposed to edge-based methods which often have to deal with dozens of small fragments, so region-based detection has performed well.

Initially, researchers relied on hand-drawn shape models [5, 27] or hand-annotated contours [10] and concentrated solely on the detection task. Since then, much research has gone into learning shape models from weakly annotated data, using only bounding box annotations and inferring the shape common to all annotated examples [7, 30]. Learned models include deformable contours [5] and combinations of several salient contour fragments [30].

In terms of hypothesis generation, the state of the art is dominated by sliding window methods, essentially trying out all possible object locations and sizes [4, 10, 22, 30]. While such methods guarantee that each object will be examined in a hypothesis verification step, they are computationally expensive and considered by some to be inelegant [8]. Several voting schemes have been explored for generating hypotheses, including Hough voting with Geometric Blur features [17], regions [8, 27] and even contour fragments [10, 18, 21], although most of these rely on contour annotation during training. The method by Opelt *et al.* [18] avoids contour annotation. Like our approach, it generates hypotheses (using voting) and defines a measure of discriminative power of a fragment but, unlike our work, it is based on a codebook of contour fragments instead of nearest neighbours in exemplar space.

In recent years, nearest neighbour methods have achieved competitive results on a variety of recognition and detection tasks, when combined with local image descriptors [2, 13, 26, 29] and keypoints [23]. Many early contour classification methods developed for matching complete contours are based on descriptors such as the shape context [1] and nearest neighbour classifiers. However, there is not much work on applying this concept to incomplete and cluttered contour fragments. Typically, chamfer matching is used in such cases to deal with incomplete contours, noise and clutter. In this paper, we build on the idea that certain parts of the image are salient and attract attention, which has been proven to be useful in general object recognition [25]. We then define a measure for finding a set of highly discriminative "trigger" fragments which are then employed to generate reliable hypotheses.

There has been great progress in the past few years, with very high detection rates on the ETHZ dataset [5]. However, we note that the majority of successful methods in recent years have relied on sliding windows, classifiers which are expensive to learn, or highly optimised chamfer matching implementations. Recent successful methods based on voting tend to use more powerful features and are usually not built on top of contour fragments. To our knowledge, no authors have applied nearest neighbour lookups to incomplete contour fragments in order to generate and verify hypotheses or to create shape models. Our paper explores these possibilities and offers a simple and fast framework for contour-based shape detection with nearest neighbours.

3 Method

We begin by defining a few important criteria for dealing with contour fragments and justify them. Let $\mathbf{s}$ be a descriptor in some high-dimensional space of an edge fragment representing part of an object contour. In Bayesian terms, this fragment was likely to be generated by some class $c \in C$ if the conditional likelihood $P(\mathbf{s}|c)$ is greater for c than that for any other class, $P(\mathbf{s}|c' \neq c)$, including the background class. $P(\mathbf{s}|c)$ can be estimated non-

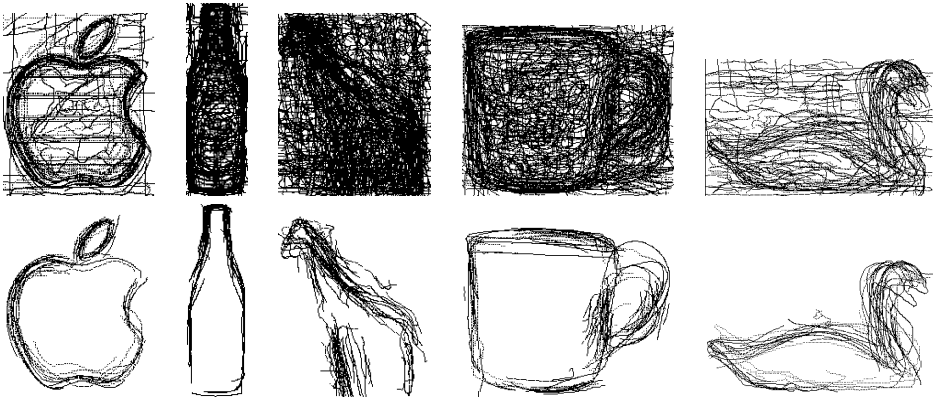


Figure 2: Model creation for the five classes of the ETHZ dataset. From left to right: Apple logo, bottle, giraffe, mug and swan. Top row: all fragments belonging to the class are superimposed after normalising the size and aspect ratio for each class. Bottom row: top 5% of all fragments for each class, after sorting them by relevance. The resulting models are ensembles of partly overlapping fragments which capture the essence of each shape, while eliminating clutter. Learning the models takes less than a minute.

parametrically, for example by using Gaussian kernels associated with some nearby samples in the feature space. In [2] it was shown that a good approximation can be obtained by using only the nearest sample:

$$P(\mathbf{s}|c) \approx \exp\left(-\frac{\|\mathbf{s}-NN_c(\mathbf{s})\|^2}{\sigma^2}\right), \tag{1}$$

where $\|\mathbf{s}-NN_c(\mathbf{s})\|$ is the distance to the nearest neighbour of $\mathbf{s}$ belonging to class c. Therefore, the log-likelihood of $P(\mathbf{s}|c)$ is proportional to the square of this distance:

$$\log P(\mathbf{s}|c) \propto -\|\mathbf{s}-NN_c(\mathbf{s})\|^2 . \tag{2}$$

A fragment is *discriminative* if it is much more likely to belong to a class c than any other class $c' \neq c$. We thus define the discriminative power of a fragment by:

$$d(\mathbf{s}) := \log \frac{P(\mathbf{s}|c)}{P(\mathbf{s}|c' \neq c)} . \tag{3}$$

As proposed in [13], we can approximate the second likelihood using the distance to the nearest sample of any other class:

$$d(\mathbf{s}) := \|\mathbf{s}-NN_{c'}(\mathbf{s})\|^2 - \|\mathbf{s}-NN_c(\mathbf{s})\|^2; \qquad c' \neq c , \tag{4}$$

which is a simple, distance-based criterion. Discriminative power will be used to select meaningful fragments for generating hypotheses in novel images.

We further define the *relevance* $r(\mathbf{s},c)$ of a fragment for class c as the probability that a similar fragment $\mathbf{s}'$ appears in an annotated sub-image I_c containing c. We consider a fragment $\mathbf{s}'$ similar to $\mathbf{s}$ if the distance between the two is less than some threshold T:

$$r(\mathbf{s},c) = P(\|\mathbf{s}-\mathbf{s}'\| < T|c); \qquad \text{where } \mathbf{s}' = NN_c(\mathbf{s}). \tag{5}$$

We estimate T by using the nearest neighbour of $\mathbf{s}$ in all training images *not* containing c. The probability in Eqn. 5 can thus be approximated by counting the number of different annotated sub-images of c in the training set, in which there is at least one fragment closer to $\mathbf{s}$ than T:

$$r(\mathbf{s},c) := \frac{\sum_{I_c} \phi(\mathbf{s},I_c)}{N_c}\,, \tag{6}$$

where N_c is the number of annotated sub-images containing c, and $\phi(\mathbf{s},c)$ is defined as:

$$\phi(\mathbf{s},I_c) = \begin{cases} 1 & \text{if } \|\mathbf{s} - NN_{I_c}(\mathbf{s})\| < T \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

Intuitively, fragments with high discriminative power $d(\mathbf{s})$ are likely to be caused by one particular class and unlikely to be caused by any other class. This makes them good candidates for generating hypotheses. On the other hand, fragments with high relevance $r(\mathbf{s},c)$ are those which appear in many images containing class c. This makes them a good choice for building shape models to be used for verifying hypotheses.

In the following sections, we describe the individual steps of the algorithm.

3.1 Feature extraction

Nearest neighbour models work best with a lot of data. Our algorithm works well if we simply extract all possible fragments longer than some minimum length and rely on efficient approximate nearest neighbour lookups [13]. However, equivalent results can be obtained faster with only a fraction of the data. In our current implementation, we extract all maximally long contiguous edge fragments from the image, by edge-linking the Berkeley edges provided with the ETHZ dataset. We extend this set by adding subsets of all fragments with one half and one quarter length shifted to cover different parts of each fragment, which strikes a good balance between the number of fragments and sufficient statistics.

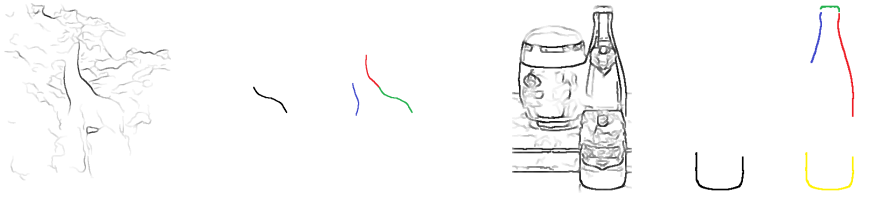
Each fragment obtained is stored as a vector of 2D coordinates, and a shape context $\mathbf{s}$ is computed for each vector, using standard parameters. When calculating the shape context, we normalise the size of each segment to make the descriptor (and thus matching) scale-invariant. We also store the location of the centre pixel of the fragment, and the offsets to the top-left and bottom-right corners of the corresponding bounding box, plus the corresponding class, for later use. For background fragments, we store the offsets to the image edges. All shape context descriptors are stored in a k-d tree for quick nearest neighbour lookups. Feature extraction takes about a minute and a half for the complete ETHZ dataset (255 images).

3.2 Model creation

Models are created from the training set only once before evaluating test images. First, we scale all training examples to a standard size and aspect ratio (see Fig. 2). In this step, we extend the shape context descriptors with the normalised 2D offsets $\mathbf{x}(\mathbf{s}) = [x,y]$ from the bounding box centre as done with SIFT descriptors in [2]:

$$\mathbf{s}^* = [\mathbf{s}, \alpha\mathbf{x}(\mathbf{s})]\,, \tag{8}$$

where α is the scaling parameter. This is the only crucial parameter in our algorithm. Luckily, near-optimal results can be obtained by choosing α such that the contribution of the



(a) (b)

Figure 3: Hypothesis generation and refinement for (a) a giraffe and (b) a bottle. The leftmost of the three images shows the edge map. The middle image shows the highly discriminative "trigger" fragment used to create an initial hypothesis. The right image shows several segments inside this hypothesis selected as being similar to model fragments from the correct class. A new hypothesis is generated such that it fits all the segments, thus refining the original one. Best seen in colour.

shape context part is roughly two times the contribution of the position part, and performance remains good for a wide range of α.

Then relevance $r(\mathbf{s},c)$ is calculated for each fragment belonging to each class c and they are sorted in descending order. We keep the most relevant 5% of the fragments and discard the rest. The resulting models consist of around 500 fragments per class. They are surprisingly clear of clutter, while representing the essence of the shape of each class (see Fig. 2). The redundancy of the ensemble of selected fragments is important for dealing with the cluttered and broken nature of fragments in the test images. The model-learning step takes less than a minute, which is very fast compared to other reported times (e.g. in the order of a few hours [30]).

3.3 Hypothesis generation

Hypothesis generation begins by extracting all contour fragments from a test image. For each fragment $\mathbf{s}$, we determine the discriminative power $d(\mathbf{s})$. In this step, only the appearance of a fragment matters, so we set $\alpha = 0$. We keep the top 20% of most discriminative fragments and discard the rest.

For each of the remaining fragments (called "trigger fragments"), we generate a hypothesis by projecting the bounding box of the nearest neighbour of $\mathbf{s}$ into a new image, scaling it to fit the position and scale of $\mathbf{s}$. Hypotheses are clustered around few promising spots, and after removing obvious overlaps, we are left with about 20 hypotheses per class per image.

Each of these hypotheses is generated using only one training example, and therefore they are not reliable. In order to find the bounding box which best characterises all fragments corresponding to the detected object, we begin by collecting fragments inside this new hypothesis. Since many of them are bound to be caused by clutter, we only keep half of the fragments which are closest to a training fragment from the correct class. This gives a set of fragments $\mathbf{s}_i$, with $i = \{1 \dots N\}$, where $\mathbf{s}_1$ represents the trigger fragment.

We can now formulate a system of linear equations involving the centre position x_i^c, y_i^c and size w_i, h_i of the bounding box, as well as the location x_i^s, y_i^s and the normalised offset ox_i^s, oy_i^s of the fragment $\mathbf{s}_i$ from the bounding box centre:

$$x_i^s = x_i^c + w_i \cdot ox_i^s \tag{9}$$

$$y_i^s = y_i^c + h_i \cdot oy_i^s \tag{10}$$

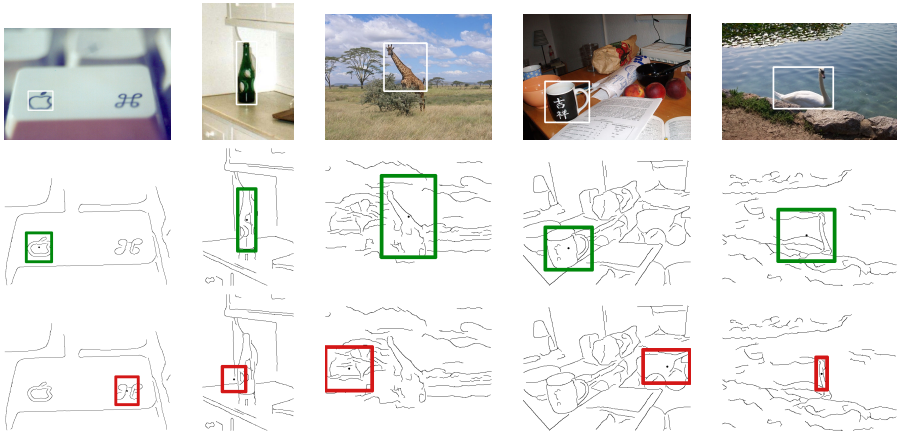


Figure 4: Example results on the ETHZ dataset. Top row: test images with ground truth annotations. Middle row: correct detections. Note the good overlap with the ground truth and the incomplete nature of edge fragments. Bottom row: some false positives. We detect all classes in all images, leading to the false bottle detection in the rightmost image.

For $N > 2$, the system is overdetermined and can be solved for x_i^c, y_i^c, w_i and h_i in a least-squares sense. Figure 3 shows two examples of trigger segments and additional segments used for bounding box regression.

3.4 Hypothesis verification

The final step in our algorithm is the verification of each hypothesis. A score for a hypothesis can be obtained from the average distance between $\mathbf{s}_i$ and fragments in the model of the corresponding class, but we have to be careful. Matching a hypothesis to the model will include the influence of clutter fragments and ignore missing parts of the shape. Both issues can be avoided by matching the fragments $\mathbf{q}$ from the model to the segments in the hypothesis in order to obtain the distance. Clutter fragments in the hypothesis will tend to be ignored because they do not resemble the models, and parts of the model contour that do not have a close match will increase the overall distance, thus penalising incomplete contours.

We define the final scoring function of a hypothesis h in the following way:

$$\text{score}(h) = \frac{1}{M} \sum_m (\mathbf{q}_m - NN_h(\mathbf{q}_m))^2 \,, \tag{11}$$

where $\mathbf{q}_m$ is the position-enhanced shape context descriptor of the m^{th} fragment in the model and $NN_h(\mathbf{q}_m)$ is its nearest neighbour in the hypothesis.

4 Evaluation

We evaluated our algorithm on the popular ETHZ shape dataset which consists of 255 images containing objects of five classes that are best defined by their shape: Apple logos, bottles, giraffes, mugs and swans. The objects exhibit large variations in scale, and there are images with multiple objects. We follow the standard evaluation procedure for this dataset and

Table 1: Comparison with the state of the art using only contour fragments. The two numbers represent the recall (in percent) for 0.3 and 0.4 false positives per image.

Method	Applelogos	Bottles	Giraffes	Mugs	Swans	**Mean**
[30]	95/95	100/100	91.3/91.3	96.7/96.7	100/100	96.5/96.5
[22]	95/95	100/100	87.2/89.6	93.6/93.6	100/100	95.2/95.6
Our	**95/95**	**96.4/100**	**81.3/85.4**	**87.1/87.1**	**88.2/94.1**	**89.6/92.3**
[19]	93.3/93.3	97/97	79.2/81.9	84.6/86.3	92.6/92.6	89.3/90.5
[17]	95/95	89.3/89.3	70.5/75.4	87.3/90.3	94.1/94.1	87.2/88.8
[7]	77.7/83.2	79.8/81.6	39.9/44.5	75.1/80	63.2/70.5	67.1/72

report the recall at 0.3 and 0.4 false positives per image (FPPI), as well as the mean across all classes. We use the Pascal criterion (intersection over union > 0.5). Although the ETHZ dataset does not contain images with instances of different classes, our algorithm has no such limitation. In our evaluation, we attempt to detect all classes on all images, which accounts on most of our false positive detections.

We are primarily interested in a comparison with methods based on contour fragments, where best methods use sliding windows and brute-force chamfer matching, and with voting methods which generate hypotheses, where all tested methods rely on stronger features than contour fragments. Our method is interesting in that it is the only competitive method which combines both aspects.

We only compare with methods which learn shape models using only bounding boxes instead of more precise additional annotations.

4.1 Detection

Table 1 shows a comparison of our method to the state of the art. We have limited the comparison to algorithms which automatically learn shape models only from bounding boxes. Hybrid methods like [27] and [10] also achieve competitive results, but they make use of hand-drawn models and manual contour annotation, respectively. Other methods like [4, 11, 14] achieve competitive results using more complex features such as HOG and region descriptors. Focusing on model-learning methods based on incomplete contour fragments, our method is only outperformed by two methods, both of which use a sliding window approach and a powerful SVM classifier, as opposed to our hypothesis-based scheme using nearest neighbour lookups.

Table 2 shows a comparison with other hypothesis-generating methods. Our method is on par with [11], which uses Geometric Blur and max-margin Hough voting. The best performing method [27] uses a combination of region segmentation, hand-drawn models, and separate SVM-based re-ranking step. We emphasise that we only make use of automatically extracted contour fragments and bounding box annotations, which makes the problem more challenging.

4.2 Effect of bounding box regression

Our method generates one hypothesis for each trigger fragment by using the scaled bounding box belonging to the nearest neighbour of that fragment. In practice, this always achieves perfect recall under the Pascal criterion, but the overlap with the detected object can often be improved. In Table 3 we evaluate the effect of the bounding box regression step in which we

Table 2: Comparison with all voting-based methods.

Method	Applelogos	Bottles	Giraffes	Mugs	Swans	**Mean**
[28]	100/100	96/97	86/91	90/91	98/100	94/96
[11]	95/95	92.9/96.4	89.6/89.6	93.6/96.7	88.2/88.2	91.9/93.2
Our	**95/95**	**96.4/100**	**81.3/85.4**	**87.1/87.1**	**88.2/94.1**	**89.6/92.3**
[19]	93.3/93.3	97/97	79.2/81.9	84.6/86.3	92.6/92.6	89.3/90.5
[27]	95.5/95.5	96.4/96.4	81.3/84.6	75.8/78.8	97/97	89.2/90.5
[17]	95/95	89.3/89.3	70.5/75.4	87.3/90.3	94.1/94.1	87.2/88.8
[7]	77.7/83.2	79.8/81.6	39.9/44.5	75.1/80	63.2/70.5	67.1/72

refine the bounding box by jointly optimising for several fragments at once (see also Fig. 3). We compare the overlap of the best-fitting hypothesis with the ground truth before and after the bounding box regression step (using the Pascal criterion). The adjusted bounding boxes are noticeably better in all cases, especially for classes with a large variation in aspect ratio (bottles and mugs).

5 Discussion

We note that there are several difficult aspects in shape detection in natural images: i) model creation, ii) hypothesis generation, and iii) hypothesis verification/classification. While there are methods which achieve slightly better results, many of them use sliding windows instead of hypothesis generation [4, 10, 22, 30], some use more complex features [17, 27], some rely on hand-drawn models instead of learning shapes [6, 27], and some of them use additional annotations [10]. Other methods use combinations of multiple classifiers.

In our work, we tackled all of these problems in a unified manner, by using the distance in the descriptor space as a simple probabilistic model for solving all three sub-problems. Our results suggest that nearest neighbour methods, while not superior to all other methods, are a fast and simple option for robust shape detection. A combination of our approach and another state-of-the-art method, for example, for hypothesis verification, will likely improve the results further. We are currently exploring this direction. Our algorithm can be easily modified to provide rotation invariance, by normalising the orientation of each segment before calculating the descriptor. This is not needed for the ETHZ dataset and additional evaluation on a different dataset is needed to test this aspect.

Not all authors report the runtime of their algorithm, but several hours for training and several seconds per test image are considered reasonable for the ETHZ dataset. [4, 30]. Our algorithm can extract features for the entire dataset in a bit over a minute and generate shape models for all five classes in about one minute. Generating hypotheses from trigger segments takes less than half a second per image. However, bounding box regression and verification of hypotheses take most of the remaining time, so processing the entire ETHZ dataset takes about an hour on a modern PC. To the best of our knowledge, this is the fastest time reported on the ETHZ dataset for a state-of-the-art method. We believe that there is much room for optimisation of the last two steps and are currently working on making our algorithm more suitable for real-time applications.

Table 3: Effect of bounding box regression. We measure the average overlap (using the Pascal criterion) between the best-fitting hypothesis and the corresponding ground truth annotation. There is a significant improvement for all classes, in particular for bottles and mugs, which exhibit the largest variation in aspect ratio.

Method	Applelogos	Bottles	Giraffes	Mugs	Swans
Overlap before BB regression	87.45	76.67	75.16	81.84	78.75
Overlap after BB regression	89.28	81.01	78.55	87.22	81.32

6 Summary

We have presented a novel algorithm for shape detection in natural images. We extract incomplete contour fragments and compute a shape context descriptor for each. We then use nearest neighbour lookups to perform all parts of the algorithm. We do not yet perform contour linking across gaps, or apply Gestalt-like operators, which might increase the performance further.

We define a Bayesian relevance criterion for learning robust shape models, and a Bayesian measure of discriminative power to identify potential object hypotheses. We refine the generated hypotheses by solving an over-determined system of linear equations, thus jointly minimising the total error for good subsets of fragments in the original hypotheses. Each hypothesis is scored using the average distance of the model fragments to the fragments in the hypothesis. Evaluation on the challenging ETHZ dataset shows that our approach is competitive with much more complex algorithms.

We are currently investigating a combination of our approach with advanced contour extraction methods and more powerful classifiers. We believe that contour linking and a more powerful verification step can significantly boost the results. There is also potential for further optimisation, making the algorithm more suitable for real-time applications. Finally, we would like to explore using prior information from scene models to improve detection [16, 24].

Acknowledgements This work was supported by the EU under the FP-7 grant ICT-2009.2.1-270247 *NeuralDynamics* and by the FCT under the grants LarSYS UID/EEA/50009/2013 and SparseCoding EXPL/EEI-SII/1982/2013.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE T-PAMI*, 24(4):509–522, 2002.

[2] O. Boiman, E. Shechtman, and M. Irani. In defense of nearest-neighbor based image classification. In *CVPR*, Anchorage, 2008.

[3] J. Canny. A computational approach to edge detection. *IEEE T-PAMI*, (6):679–698, 1986.

[4] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE T-PAMI*, 32(9):1627–1645, 2010.

[5] V. Ferrari, T. Tuytelaars, and L. Van Gool. Object detection by contour segment networks. *ECCV*, (4):14–28, 2006.

[6] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *IEEE T-PAMI*, 30(1):36–51, 2008.

[7] V. Ferrari, F. Jurie, and C. Schmid. From images to shape models for object detection. *IJCV*, 87(3):284–303, July 2009.

[8] C.H. Gu, J.J. Lim, P. Arbelaez, and J. Malik. Recognition using regions. In *CVPR*, pages 1030–1037, 2009.

[9] M. Liu, O. Tuzel, A. Veeraraghavan, and R. Chellappa. Fast directional chamfer matching. In *CVPR*, pages 1696–1703, 2010.

[10] T. Ma and L. J. Latecki. From partial shape matching through local deformation to robust global shape similarity for object detection. In *CVPR*, pages 1441–1448, 2011.

[11] S. Maji and J. Malik. Object detection using a max-margin hough transform. In *CVPR*, pages 1038–1045, 2009.

[12] D.R. Martin, C.C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE T-PAMI*, 26(5):530–49, 2004.

[13] S. McCann and D.G. Lowe. Local naive bayes nearest neighbor for image classification. In *CVPR*, pages 3650–3656, Providence, 2012.

[14] A. Monroy, A. Eigenstetter, and B. Ommer. Beyond straight lines - object detection using curvature. In *ICIP*, pages 3561–3564, 2011.

[15] M. Muja and D. G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISSAPP*, pages 331–340. INSTICC Press, 2009.

[16] B. Neumann and K. Terzić. Context-based probabilistic scene interpretation. In *IFIP AI*, pages 155–164, Brisbane, Sep 2010.

[17] B. Ommer and J. Malik. Multi-scale object detection by clustering lines. In *ICCV*, pages 484–491, 2009.

[18] A. Opelt, A. Pinz, and A. Zisserman. A boundary-fragment-model for object detection. In *ECCV*, 2006.

[19] H. Riemenschneider, M. Donoser, and H. Bischof. Using partial edge contour matches for efficient object category localization. In *ECCV*, pages 29–42. 2010.

[20] K. Schindler and D. Suter. Object detection by global contour shape. *Pattern Recognition*, 41(12):3736–3748, December 2008.

[21] J. Shotton, A. Blake, and R. Cipolla. Contour-based learning for object detection. In *ICCV*, pages 503–510, 2005.

[22] P. Srinivasan, Q. Zhu, and J. Shi. Many-to-one contour matching for describing and discriminating object shape. In *CVPR*, pages 1673–1680, 2010.

[23] K. Terzić and J.M.H. du Buf. An efficient naive bayes approach to category-level object detection. In *ICIP*, pages 1658–1662, Paris, 2014.

[24] K. Terzić, L. Hotz, and J. Šochman. Interpreting structures in man-made scenes: Combining low-level and high-level structure sources. In *ICAART 2010*, Valencia, Spain, January 2010.

[25] K. Terzić, J.M.F. Rodrigues, and J.M.H. du Buf. Fast cortical keypoints for real-time object recognition. In *ICIP*, pages 3372–3376, Melbourne, Sep 2013.

[26] R. Timofte, T. Tuytelaars, and L. J. Van Gool. Naive bayes image classification: Beyond nearest neighbors. In *ACCV*, pages 689–703, 2012.

[27] A. Toshev, B. Taskar, and K. Daniilidis. Object detection via boundary structure segmentation. In *CVPR*, pages 950–957, 2010.

[28] A. Toshev, B. Taskar, and K. Daniilidis. Shape-based object detection via boundary structure segmentation. *IJCV*, 2012.

[29] T. Tuytelaars, M. Fritz, K. Saenko, and T. Darrell. The NBNN kernel. In *ICCV*, pages 1824–1831, Barcelona, Nov 2011.

[30] P. Yarlagadda and B. Ommer. From meaningful contours to discriminative object shape. *ECCV*, pages 766–779, 2012.