From Contours to Regions: An Empirical Evaluation

Anonymous CVPR submission

Paper ID 1134

Abstract

We propose an algorithm that constructs a hierarchy of regions from the output of any given contour detector. This method provides a powerful generic grouping mechanism when applied to the output of a high-performance contour detector. The primary focus of this paper is on experimental evaluation. For this purpose, we examine a number of different evaluation metrics and standard segmentation datasets in detail. We also benchmark several published segmentation algorithms which are in common use. The consensus results of these varied benchmarks suggest that the method proposed here provides state-of-the-art segments and contours.

1. Introduction

Contour detection, image segmentation, and perceptual grouping lie at the heart of computer vision, reducing the complexity of an image with millions of pixels to a small number of cohesive entities suitable for high-level analysis. For example, contours convey important information about shape and object identity which has been exploited successfully in detection and recognition. While extracting contours from local image features is an old problem, there have been significant advances in contour detection in the last few years [13, 30, 22, 9, 39]. This is best summarized by the boundary benchmark shown in Figure 1 which is based on the quantitative evaluation methodology proposed in [22], measuring the precision and recall of a contour detector with respect to human-marked boundaries.

Despite these successes, contours are not always enough. For many applications such as recognition, a segmentation of the image into closed regions is far more natural. Segments have the advantage of automatically providing spatial support and scale estimation for the objects in the image. Applications utilizing segments typically exploit one of a handful of generic segmentation methods such as [8, 12, 7]. However, these segmentation algorithms often fail to deliver the desired segmentation without careful tuning of parameters and choice of the number of segments in a given image. As a consequence, region-based attempts to recognition have been done with inaccurate segmentations, often relying on one of the following strategies: (1) an over-segmentation [32, 16], where the image domain is partitioned into a large number of "super-pixels", subsequently assembled using object-specific knowledge; (2) a large number of candidate segments, collected from different sets of parameters from one or several segmentation algorithms [28, 20]; (3) transitive closure of a segmentation tree [1].

It also appears to be difficult to define a single criterion for benchmarking segmentations which is robust to variability in the number of segments while remaining sensitive to the segmentation quality. The lack of a commonly agreed upon measure of segmentation performance has had a limiting effect on research progress in this area.

In this paper, we propose an alternate solution to some of these difficulties. We describe an algorithm that, starting from the output of a contour detector, constructs a hierarchy of segmentations. Our method, based on a variant of the venerable watershed [4], can be seen as generic machinery for going from contours to hierarchical regions. Contours encoded in the hierarchical segmentation retain real-valued weights indicating their likelihood of being a true boundary. For a given threshold, the output is a set of closed contours that can be treated as either a segmentation or as a boundary detector for the purposes of benchmarking.

To establish the value of this technique, we examine a number of different evaluation metrics and standard datasets for both boundary and region detection. We also evaluate several other publicly available segmentation algorithms. Based on this extensive testing, we report two important empirical results:

1. Weighted boundary maps can be converted into hierarchical segmentations without loss of boundary precision or recall.

2. Using the high-performance gPb contour detector [18] as input, our method provides a powerful mid-level grouping mechanism which outperforms existing segmentation algorithms.

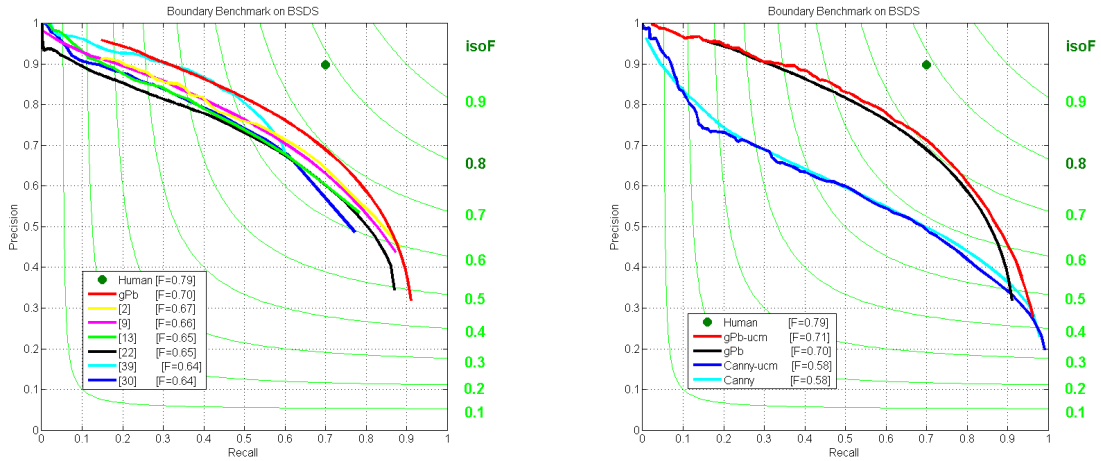

Figure 1. **Boundary benchmark on the Berkeley dataset [14]. Left: Contours.** Leading approaches to contour detection are ranked according to their F-measure (harmonic mean of precision and recall) with respect to human ground-truth boundaries. Iso-F curves are shown in green. The leading approach, gPb [18], does not produce closed boundaries required to form a segmentation. **Right: Hierarchical Regions.** Our algorithm produces a hierarchical segmentation from the output of any contour detector. Benchmarking the resulting segment boundaries (UCM) shows that our method constructs regions without losing performance on the boundary benchmark. In fact, we obtain a boost in performance when using the gPb detector as input.

2. Prior Work

While contours and segments are closely related (equivalent in the case of closed, non-self-intersecting contours), researchers have largely pursued the sub-problems of contour detection and segmentation independently.

Common approaches to segmentation involve integrating features such as color or texture over local patches of the image and then clustering those feature vectors based on, e.g., fitting mixture models [3, 38], mode-finding [7], or graph partitioning [33, 5, 19, 12]. One challenge that such clustering approaches must face is that smooth changes in texture or brightness due to perspective or shading can cause distant patches on the same surface to appear dissimilar despite belonging to the same image region, creating a tendency toward *over-segmentation*.

Contour based approaches ignore smooth variations by directly searching for locations in the image where brightness or other features undergo rapid local changes [6, 27]. These high-gradient edge fragments can then be linked together in order to identify extended, smooth contours [26, 37, 10, 31]. However, in the contour based approach, there remains the problem of producing segments from non-closed boundary fragments. Without some mechanism for enforcing closure, a segmentation built up from locally detected contours will mistakenly join regions due to "leaks" in the bounding contour and tend to result in an *under-segmentation*.

There have been several proposals for including contour information with segmentation in a single framework, such as the variational formulations suggested by [25, 24] or the discrete algorithms of [21, 17]. The utility of integrating both edge and region cues in order to segment images of natural scenes has been quantified statistically [15] and practically demonstrated in full segmentation systems [35, 19].

While these unified approaches integrate contour information, they still operate on a segment based representation. Contours have the advantage that it is fairly straightforward to represent uncertainty in the presence of a true underlying contour, i.e. by associating a binary random variable to it. It is not immediately obvious how to represent uncertainty about a segmentation. One possibility, which we exploit here, is the Ultrametric Contour Map (UCM) [2] which defines a duality between closed, non-self-intersecting weighted contours and a hierarchy of segmentations. We show that making this shift in representation from a single segmentation to a nested collection of segmentations allows us to garner benefits from both approaches.

3. Hierarchical Regions

We consider a contour detector, whose output $pb(x, y, \theta)$ predicts the probability of an image boundary at location (x, y) and orientation θ. We build our hierarchical regions by exploiting the information in this contour signal. For best results, we employ the high-performance detector gPb [18], but any source of such a signal, *e.g.* the Canny edge detector before thresholding, can be used.

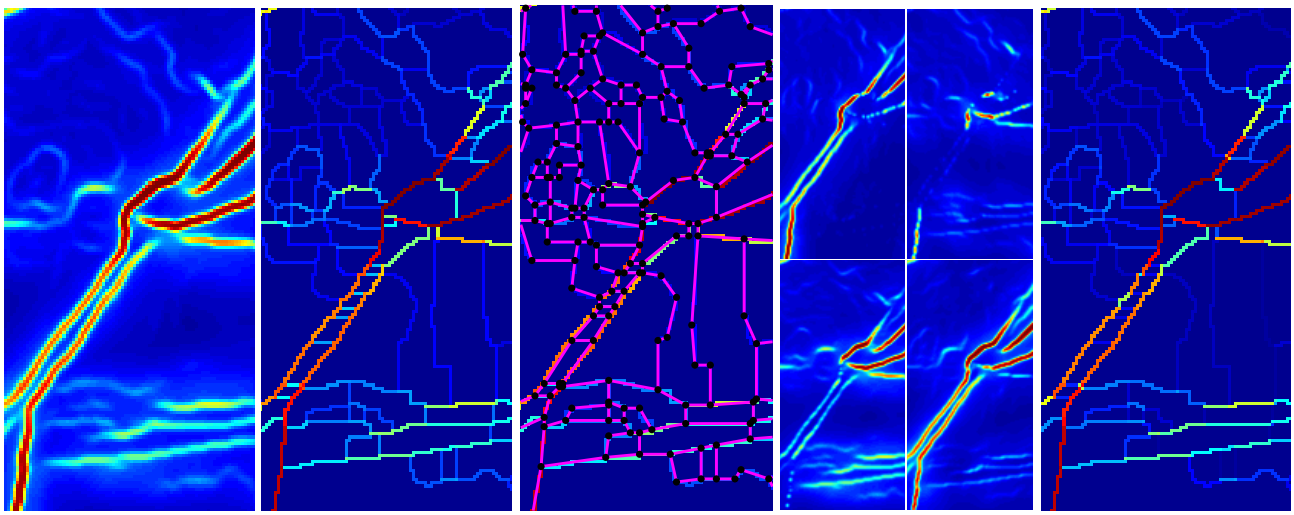

Figure 2. **Transferring boundary strength onto watershed arcs. Left:** Input boundary signal $pb(x, y)$. **Middle Left:** Watershed arcs computed from $pb(x, y)$. Note that thin regions give rise to artifacts. **Middle:** Watershed arcs with an approximating straight line segment subdivision overlaid. We compute this subdivision in a scale-invariant manner by recursively breaking an arc at the point maximally distant from the straight line segment connecting its endpoints. Subdivision terminates when the distance from the line segment to every point on the arc is less than a fixed fraction of the segment length. **Middle Right:** Oriented boundary strength $pb(x, y, \theta)$ for four orientations θ. In practice, we sample eight orientations. **Right:** Watershed arcs reweighted according to pb at the orientation of their associated line segments. Artifacts are suppressed as their orientations do not agree with the underlying $pb(x, y, \theta)$ signal.

First, we construct a finest partition for the hierarchy, an over-segmentation whose regions determine the highest level of detail considered. This is done by computing the maximal response of the pb operator over orientations. Then, following the tradition in morphology, we consider as candidate locations for homogeneous regions the regional minima of $pb(x, y)$ and apply the watershed transform [4] on this topographical surface. As a result, the catchment basins of the minima, noted $\mathcal{P}_0$, provide the regions of the finest partition and the corresponding watershed arcs, noted $\mathcal{K}_0$, the possible locations of the boundaries.

Next, we transfer the strength of the boundaries, given by the probability $pb(x, y, \theta)$, to the locations $\mathcal{K}_0$. For this purpose, we approximate the watershed arcs with line segments, and weight each point in $\mathcal{K}_0$ by the $pb(x, y, \theta)$ value at that point, in the direction θ given by the orientation of the corresponding line segment, as detailed in Figure 2. This procedure enforces consistency between the strength of the boundaries of $\mathcal{K}_0$ and the underlying pb signal and serves to remove possible artifacts of the watershed algorithm.

The hierarchy itself is constructed by a graph-based region merging algorithm. An initial graph is defined, where the nodes are the regions in $\mathcal{P}_0$, and the links join adjacent regions and are weighted by a measure of similarity between regions. The algorithm proceeds by sorting the links by similarity and iteratively merging the regions. This process produces a tree of regions, where the leaves are the elements of $\mathcal{P}_0$, the root is the entire image domain and the regions are ordered by the inclusion relation.

In our case, we use as similarity between two adjacent regions the average strength of their common boundary in $\mathcal{K}_0$. Since this value cannot decrease during the merging process, the above algorithm is guaranteed to produce an ultrametric distance on $\mathcal{P}_0 \times \mathcal{P}_0$ [2]. As a consequence, the constructed region tree has the structure of indexed hierarchy and can be described by a dendrogram, where the height of each node is the value of the similarity at which it first appears. Furthermore, the whole hierarchy can in this case be represented as an Ultrametric Contour Map (UCM), the real-valued image obtained by weighting each boundary between two regions by its scale of disappearance.

Figure 3 presents an example of our method. The UCM is a weighted contour image that, by construction, has the remarkable property of producing a set of closed curves for any threshold. Conversely, it is a convenient representation of the region tree since the segmentation at a scale k can be easily retrieved by thresholding the UCM at level k. In our case, since our notion of *segmentation scale* is the average contour strength, our UCM values reflect a notion of contrast between neighboring regions. We explore the use of two different contour detectors in constructing the UCM, Canny without thresholding and gPb, and refer to the resulting hierarchical segmentations as Canny-ucm and gPb-ucm, respectively. Figure 4 illustrates results of our state-of-the-art gPb-ucm algorithm on images selected from the Berkeley Segmentation dataset [14].

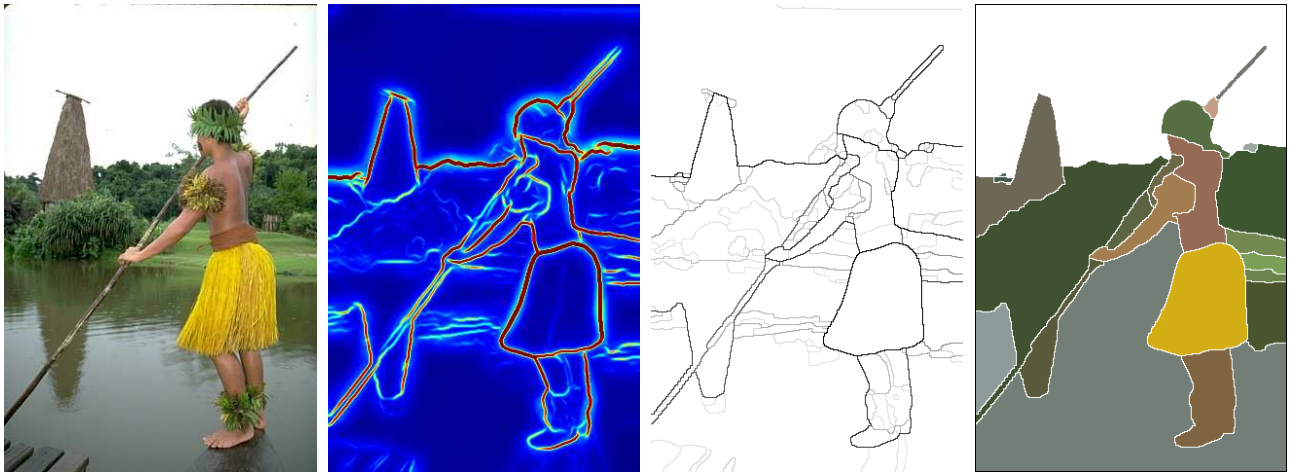

Figure 3. **Overview of our approach. Left:** Original image. **Middle Left:** Maximal response of contour detector gPb over orientations. **Middle Right:** Ultrametric Contour Map (UCM) produced by our system. **Right:** Segmentation obtained by thresholding the UCM at level 0.4, with segments represented by their mean color.

4. Other Segmentation Methods

To provide a basis of comparison for the performance of our segmentation method, we make use of the publicly available implementations of the other segmentation methods described below.

4.1. Felzenszwalb and Huttenlocher (Felz-Hutt)[12]

The region-merging approach advocated by Felzenszwalb and Huttenlocher [12] has been widely used by the computer vision community. This algorithm attempts to partition image pixels into components such that the resulting segmentation is neither too coarse nor too fine. Given a graph in which pixels are nodes and edge weights measure the dissimilarity between nodes (e.g. color differences), each node is initially placed in its own component. Define the internal difference of a component $Int(C)$ as the largest weight in the minimum spanning tree of C. Considering edges in non-decreasing order by weight, each step of the algorithm merges components C_1 and C_2 connected by the current edge if the edge weight (equivalently difference between components) is less than

$$min(Int(C_1) + \tau(C_1), Int(C_2) + \tau(C_2)) \qquad (1)$$

where $\tau(C) = k/|C|$. k is a scale parameter that can be used to set a preference for component size. Merging stops when the difference between components exceeds the internal component difference.

4.2. Mean Shift [7]

Mean shift is another popular image segmentation technique. In [7] pixels are represented in the joint spatial-range domain by concatenating their spatial coordinates and color values into a single vector. Applying mean shift filtering in this domain yields a convergence point for each pixel. Regions are formed by grouping together all pixels whose convergence points are closer than h_s in the spatial domain and h_r in the range domain, where h_s and h_r are respective bandwidth parameters. Additional merging can also be performed to enforce a constraint on minimum region area.

4.3. Multiscale Normalized Cuts (NCuts) [8]

Graph partitioning approaches based on the Normalized Cuts criterion have also been the focus of much recent work on image segmentation. In this framework, image cues are used to define an affinity matrix W whose entries encode pixel similarity. Image segmentation is performed by partitioning the graph $G = (V, E, W)$ with pixels as nodes V and edge weights W using the generalized eigenvectors of the linear system $(D-W)x = \lambda Dx$, where $D_{ii} = \sum_j W_{ij}$.

We compare against the latest variant of this approach which exploits multiscale information [8]. The fact that W must be sparse in order to avoid a prohibitively expensive computation limits the naive approach to using only local pixel affinities. Cour *et al.* solve this limitation by computing sparse affinity matrices at multiple scales, setting up cross-scale constraints, and deriving a new eigenvalue problem for this constrained multiscale Normalized Cut.

5. Empirical Evaluation

We evaluate the performance of the segmentation algorithms discussed previously using multiple benchmarks across several datasets.

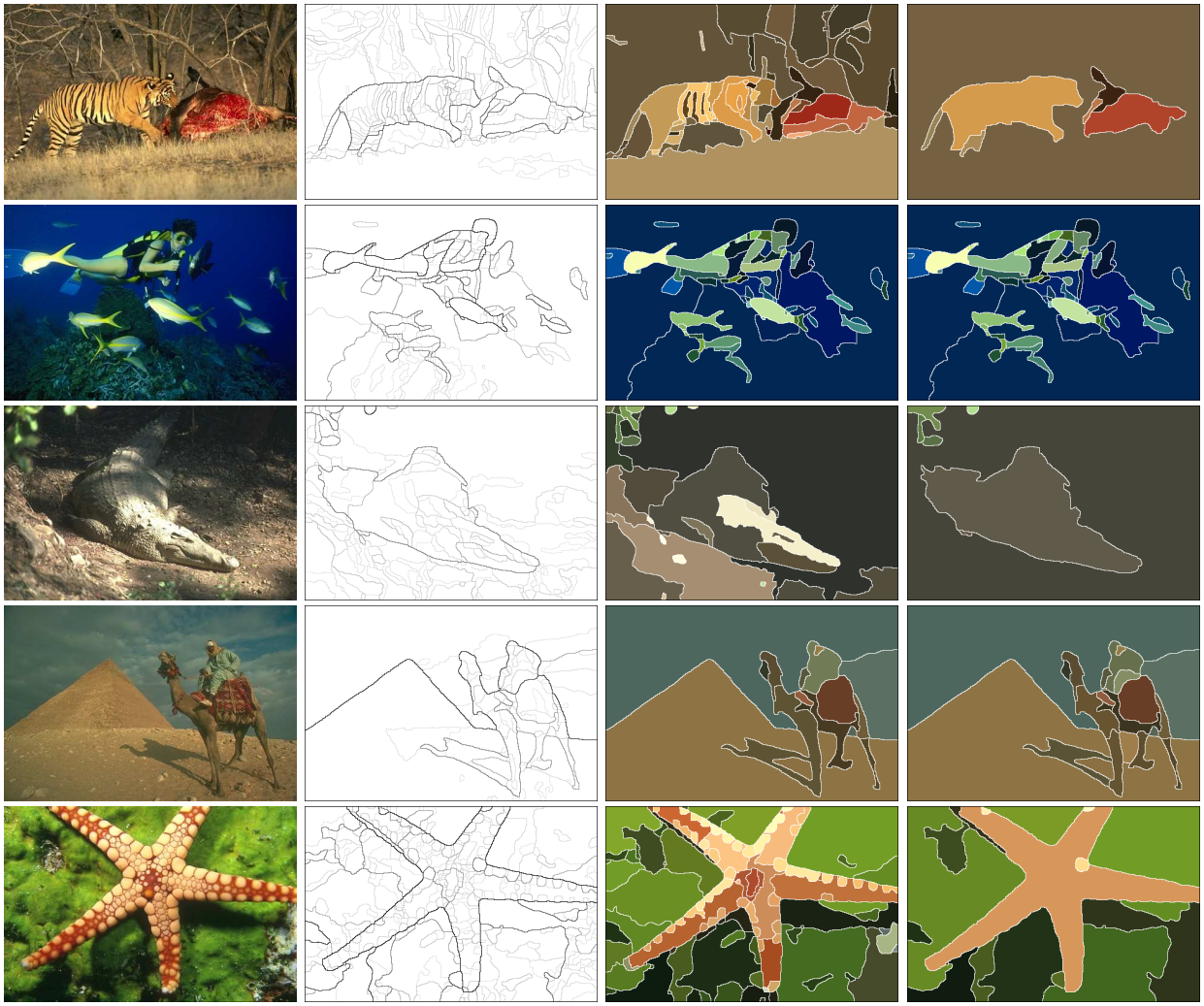

Figure 4. **Hierarchical segmentation results. From left to right:** Original image, Ultrametric Contour Map (UCM) produced by gPb-ucm, and segmentations obtained by thresholding at the optimal dataset scale (ODS) and optimal image scale (OIS).

5.1. Benchmarks

There have been thousands of papers on image segmentation. However, only recently has the community acknowledged the necessity of quantitative performance analysis in order to monitor and guide the progress of the field. A first step in that direction was the release of the Berkeley Segmentation Dataset (BSDS) [14], a database of 300 natural images, manually segmented by a number of different subjects. The ground-truth data for this large collection shows the diversity, yet high consistency, of human segmentation. We examine metrics for evaluating both boundaries and regions.

5.1.1 Precision-Recall on Boundaries

In the last years, the boundary-based evaluation methodology developed by Martin *et al*. [22] on the BSDS, has become a standard, as demonstrated by its widespread use [30, 13, 9, 2, 39, 18]. This framework considers two aspects of detection performance. Precision measures the fraction of true positives in the contours produced by a detector. Recall measures the fraction of ground-truth boundaries detected. If the algorithm provides a binary output, one obtains a point in the precision-recall plane for each image. If the output is real-valued, one obtains a curve parametrized by the threshold of the detector, quantifying its performance at all regimes. The global F-measure, defined as the harmonic mean of precision and recall, may provide a useful summary score for the algorithm.

In the experimental section, we report three different quantities for an algorithm: the **Optimal Dataset Scale (ODS)** or best F-measure on the dataset for a fixed scale, the **Optimal Image Scale (OIS)** or aggregate F-measure

on the dataset for the best scale in each image, and the **Average Precision (AP)** on the full recall range (equivalently, the area under the precision-recall curve).

This benchmarking methodology possess the appealing property that it allows the comparison of region-based segmentation and contour detection methods in the same framework. Any segmentation algorithm automatically provides contours in the form of the boundaries of the regions in the segmentation.

However, for an algorithm that produces a segmentation, a methodology that directly evaluates the quality of the segments is also desirable. Some types of errors, *e.g.* a missing pixel in the boundary between two segments, may not be reflected in the boundary benchmark, but can have substantial consequences for the quality of the segmentation, *e.g.* incorrectly merging two large segments. It can also be argued that the boundary benchmark favors contour detectors over segmentation methods, since the latter have the additional constraint of producing closed curves. We will therefore also consider various region-based metrics and now briefly review three of them.

5.1.2 Variation of Information [23]

This metric was introduced for the purpose of clustering comparison. It measures the distance between two segmentations in terms of their average conditional entropy given by

$$VI(C, C') = H(C) + H(C') - 2I(C, C') \qquad (2)$$

where H and I represent respectively the entropies and mutual information between two clusterings of data C and C'. In our case, the two clusterings are test and ground-truth segmentations. Although VI possesses some interesting theoretical properties [23], its perceptual meaning and applicability in the presence of several ground-truth segmentations remains unclear.

5.1.3 Rand Index [29]

Originally, the Rand Index [29], was introduced for general clustering evaluation. It operates by comparing the compatibility of assignments between pairs of elements in the clusters. In our case, the Rand Index between test and ground-truth segmentations S and G is given by the sum of the number of pairs of pixels that have the same label in S and G and the number that have different labels in both segmentations, divided by the total number of pairs of pixels. Variants of the Rand Index have been proposed [36, 38] for dealing with the case of multiple ground-truths when evaluating segmentations. Given a set of ground-truth segmentations $\{G_k\}$, the Probabilistic Rand Index is defined as

$$PRI(S, \{G_k\}) = \frac{1}{T} \sum_{i<j} [c_{ij} p_{ij} + (1 - c_{ij})(1 - p_{ij})] \qquad (3)$$

where c_{ij} is the event of pixels i and j having the same label and p_{ij} its probability. When the sample mean is used to estimate the p_{ij}, (3) amounts to averaging the Rand Index among different ground-truth segmentations. However, the PRI has been reported to suffer from a small dynamic range [36, 38], and its values across images and algorithms are often very similar. In [36], this drawback is addressed by a normalization with an empirical estimation of its expected value.

5.1.4 Segmentation Covering

The **overlap** between two regions R and R', defined as:

$$\mathcal{O}(R, R') = \frac{|R \cap R'|}{|R \cup R'|} \qquad (4)$$

has been used for the evaluation of the pixel-wise classification task in recognition [20, 11].

We define the **covering** of a segmentation S by a segmentation S' by:

$$\mathcal{C}(S' \to S) = \sum_{R \in S} \sum_{p \in R} \mathcal{O}(R, \arg\min_{R' \in S'} \mathcal{O}(R, R')) \qquad (5)$$

Similarly, the covering of a machine segmentation S by a family of ground truth segmentations $\{G_i\}$ is defined by first covering S separately with each human map $\{G_i\}$ in turn, and then averaging over the different humans, so that to achieve perfect covering the machine segmentation must explain all of the human data.

We can then define two quality descriptors for regions, the covering of S by $\{G_i\}$ and the covering of $\{G_i\}$ by S. Due to space constraints, we only include results for the covering of the ground-truth by S. For a family of machine segmentations $\{S_i\}$, corresponding to different scales of a hierarchical algorithm or different set of parameters, we report three different quantities: ODS, OIS, and the best possible covering of the ground-truth by segments in $\{S_i\}$.

5.2. Additional Datasets

We concentrated our experiments on the BSDS, because it is the most complete dataset available for our purposes, has been used in several publications and has the advantage of providing several human-labeled segmentations per image. In addition, we present in this section experiments on two other publicly available labeled datasets: MSRC and the segmentation subset of the PASCAL 2008 challenge. Due to time limitations, we report only the comparison between Canny-ucm and gPb-ucm for these datasets.

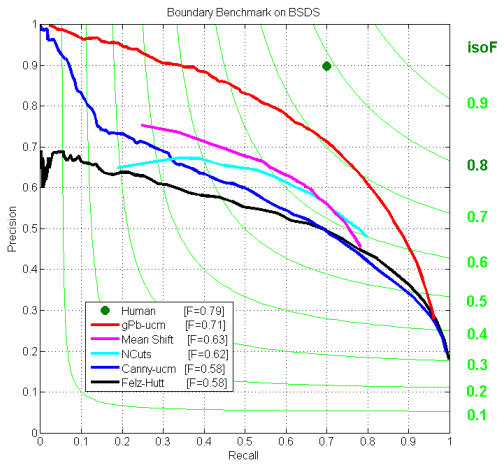

Figure 5. **Evaluating segmentation boundaries on the BSDS.** Our hierarchical segmentation algorithm gPb-ucm produces segments whose boundaries match human ground-truth better than those produced by alternative Mean Shift, Normalized Cuts, or region-merging (Felz-Hutt) segmentation approaches. The quality of the contour detector (gPb vs Canny) on which we build significantly influences the quality of the resulting segmentation.

5.2.1 MSRC [34]

We consider the MSRC object recognition database, composed of 591 natural images belonging to 21 object classes. The images are fully parsed and offer enough variety. The evaluation is performed using the ground-truth object instance labeling of [20], which is cleaner and more precise than the original data.

5.2.2 PASCAL challenge 2008 [11]

We use the train and validation sets of the segmentation task on the PASCAL challenge 2008, composed of 1023 images. This is one of the most challenging and varied datasets for recognition. We evaluate performance with respect to the object instance labels provided. However, it should be noted that only the objects belonging to the 20 categories of the challenge are labeled, and 76% of all pixels are unlabeled.

6. Results

Figure 5 and Table 1 present results for the BSDS boundary benchmark, while Figure 6 and Table 2 present region benchmarks for this dataset. The boundary benchmark appears best able to discriminate the performance of the different segmentation algorithms. Table 3 displays additional region benchmark results on the MSRC and PASCAL datasets.

Our segmentation algorithm, gPb-ucm, offers the best performance on every dataset and for every benchmark criteria we tested. In addition, our algorithm is straightforward, fast, and has no parameters to tune. We hope that our generic grouping machinery will be useful for recognition applications.

Method	ODS	OIS	AP
human	0.79	0.79	
gPb-ucm	**0.71**	**0.74**	**0.77**
Mean Shift	0.63	0.66	0.62
NCuts	0.62	0.66	0.59
Canny-ucm	0.58	0.63	0.59
Felz-Hutt	0.58	0.62	0.54
gPb	0.70	0.72	0.75
Canny	0.58	0.62	0.60

Table 1. **Boundary benchmarks on the BSDS.** We benchmark boundaries produced by five different segmentation methods (upper table) and two contour detectors (lower table) against human ground-truth. Shown are the F-measures when choosing an optimal scale for the entire dataset (ODS) or per image (OIS), as well as the average precision (AP). Figure 5 shows the full precision-recall curves for the boundaries produced by the segmentation algorithms.

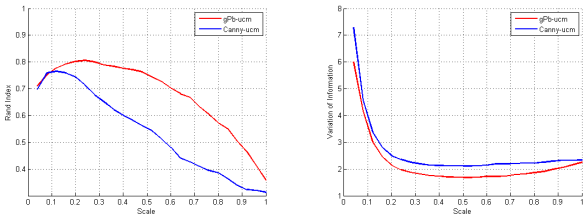

Figure 6. **Evaluating regions on the BSDS.** The influence of the contour detector on segmentation quality is also evident when benchmarking the regions of the resulting hierarchical segmentation. **Left:** Probabilistic Rand Index. **Right:** Variation of Information.

Method	ODS	OIS	best		PRI	VI
human	0.73	0.73			0.87	1.16
gPb-ucm	**0.58**	**0.64**	**0.74**		**0.81**	**1.68**
Mean Shift	0.54	0.58	0.64		0.78	1.83
Felz-Hutt	0.51	0.58	0.68		0.77	2.15
Canny-ucm	0.48	0.56	0.67		0.77	2.11
NCuts	0.44	0.53	0.66		0.75	2.18

Table 2. **Region benchmarks on the BSDS.** For each segmentation method, the leftmost three columns report the score of the best covering of ground-truth segments according to optimal dataset scale (ODS), optimal image scale (OIS), or best covering criteria. The rightmost two columns compare the segmentation methods against ground-truth using the probabilistic Rand Index (PRI) and Variation of Information (VI) benchmarks, respectively.

MSRC	ODS	OIS	best
gPb-ucm	**0.66**	**0.75**	**0.78**
Canny-ucm	0.57	0.68	0.72
PASCAL08	ODS	OIS	best
gPb-ucm	**0.45**	**0.58**	**0.61**
Canny-ucm	0.40	0.53	0.55

Table 3. **Region benchmarks on MSRC and PASCAL08.**

References

[1] N. Ahuja and S. Todorovic. Connected segmentation tree: A joint representation of region layout and hierarchy. *CVPR*, 2008. 1

[2] P. Arbeláez. Boundary extraction in natural images using ultrametric contour maps. *POCV*, 2006. 2, 3, 5

[3] S. Belongie, C. Carson, H. Greenspan, and J. Malik. Color- and texture-based image segmentation using EM and its application to content-based image retrieval. pages 675–682, 1998. 2

[4] S. Beucher and F. Meyer. *Mathematical Morphology in Image Processing*, chapter 12: The Morphological Approach to Segmentation: The Watershed Transformation, pages 433–481. Marcel Dekker, 1992. 1, 3

[5] E. Borenstein and S. Ullman. Combined top-down/bottom-up segmentation. *PAMI*, 2008. 2

[6] J. Canny. A computational approach to edge detection. *PAMI*, 1986. 2

[7] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *PAMI*, 2002. 1, 2, 4

[8] T. Cour, F. Benezit, and J. Shi. Spectral segmentation with multiscale graph decomposition. *CVPR*, 2005. 1, 4

[9] P. Dollar, Z. Tu, and S. Belongie. Supervised learning of edges and object boundaries. *CVPR*, 2006. 1, 5

[10] J. Elder and S. Zucker. Computing contour closures. *ECCV*, 1996. 2

[11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. PASCAL 2008 Results. http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html. 6, 7

[12] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based segmentation algorithm. *IJCV*, 2004. 1, 2, 4

[13] P. Felzenszwalb and D. McAllester. A min-cover approach for finding salient curves. *POCV*, 2006. 1, 5

[14] C. Fowlkes, D. Martin, and J. Malik. The Berkeley Segmentation Dataset and Benchmark (BSDB). www.cs.berkeley.edu/projects/vision/grouping/segbench/. 2, 3, 5

[15] C. Fowlkes, D. Martin, and J. Malik. Learning affinity functions for image segmentation: combining patch-based and gradient-based approaches. *CVPR*, 2003. 2

[16] D. Hoiem, A. Efros, and M. Hebert. Geometric context from a single image. *ICCV*, 2005. 1

[17] I. Jermyn and H. Ishikawa. Globally optimal regions and boundaries as minimum ratio weight cycles. *PAMI*, 2001. 2

[18] M. Maire, P. Arbelaez, C. Fowlkes, and M. Malik. Using contours to detect and localize junctions in natural images. *CVPR*, 2008. 1, 2, 5

[19] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *IJCV*, 2001. 2

[20] T. Malisiewicz and A. A. Efros. Improving spatial support for objects via multiple segmentations. *BMVC*, 2007. 1, 6, 7

[21] R. Malladi, J. Sethian, and B. Vemuri. Shape modelling with front propogation: A level set approach. *PAMI*, 1995. 2

[22] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color and texture cues. *PAMI*, 2004. 1, 5

[23] M. Meila. Comparing clusterings: An axiomatic view. *ICML*, 2005. 6

[24] J.-M. Morel and S. Solimini. *Variational Methods in Image Segmentation*. Birkhäuser, 1995. 2

[25] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions, and associated variational problems. *Comm. in Pure and Applied Math.*, pages 577–684, 1989. 2

[26] P. Parent and S. Zucker. Trace inference, curvature consistency, and curve detection. *PAMI*, 1989. 2

[27] P. Perona and J. Malik. Detecting and localizing edges composed of steps, peaks and roofs. *ICCV*, 1990. 2

[28] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. *ICCV*, 2007. 1

[29] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66:846850, 1971. 6

[30] X. Ren, C. Fowlkes, and J. Malik. Scale-invariant contour completion using conditional random fields. *ICCV*, 2005. 1, 5

[31] X. Ren, C. Fowlkes, and J. Malik. Learning probabilistic models for contour completion in natural images. *IJCV*, 2008. 2

[32] X. Ren and J. Malik. Learning a classification model for segmentation. *ICCV*, 2003. 1

[33] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 2000. 2

[34] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. *ECCV*, 2006. 7

[35] Z. Tu and S. Zhu. Image segmentation by data-driven markov chain monte carlo. *PAMI*, 2002. 2

[36] R. Unnikrishnan, C. Pantofaru, and M. Hebert. Toward objective evaluation of image segmentation algorithms. *PAMI*, 2007. 6

[37] L. Williams and D. Jacobs. Stochastic completion fields: a neural model of illusory contour shape and salience. *ICCV*, 1995. 2

[38] A. Yang, J. Wright, Y. Ma, and S. Sastry. Unsupervised segmentation of natural images via lossy data compression. *CVIU*, 2008. 2, 6

[39] Q. Zhu, G. Song, and J. Shi. Untangling cycles for contour grouping. *ICCV*, 2007. 1, 5