Rigid Shape Matching by Segmentation Averaging

Hongzhi Wang, *Member*, *IEEE*, and John Oliensis, *Senior Member*, *IEEE*

Abstract—We use segmentations to match images by shape. The new matching technique does not require point-to-point edge correspondence and is robust to small shape variations and spatial shifts. To address the unreliability of segmentations computed bottom-up, we give a closed form approximation to an average over all segmentations. Our method has many extensions, yielding new algorithms for tracking, object detection, segmentation, and edge-preserving smoothing. For segmentation, instead of a maximum a posteriori approach, we compute the "central" segmentation minimizing the average distance to all segmentations of an image. For smoothing, instead of smoothing images based on local structures, we smooth based on the global optimal image structures. Our methods for segmentation, smoothing, and object detection perform competitively, and we also show promising results in shape-based tracking.

Index Terms—Shape matching, image segmentation, mutual information.

1 Introduction

The shape of an object (as conveyed by edge curves) is among its most distinctive features. Though a category of objects can vary greatly in appearance as the illumination or coloring changes, the overall shape of the objects can be relatively invariant, see Fig. 4. People can easily recognize objects just from their shapes, and applying shape for matching and recognition has been an important topic in computer vision.

From the computational point of view, the main obstacle to matching or recognizing shapes is deformation. Since the possible deformations grows exponentially in the size of the shape, matching shapes *globally* by searching for the correct deformation is intractable in practice. The problem is further complicated by occlusion and missed edge detections.

To avoid searching globally over all deformations, one natural approach is to match shape "fragments" *locally*. Indeed, local edge fragments are easy to find matches for, but their simplicity also decreases their distinctiveness for recognition. Hence, a global integration step is required to eliminate outlier matches and establish correct correspondences of the local fragments, and for complex deforming shapes this again involves an exponential search.

To address this global/local dilemma in shape matching, some recent approaches [32], [17] use grouped edge fragments as *intermediate* representations. These grouped curves have more specificity and fewer potential matches than individual edge fragments yet are still relatively easy to match and occur often enough to survive occlusions and detection failures. However, bottom-up grouping isn't reliable. As a result, these approaches are limited to matching specific object classes; they learn distinctive fragment groups for the given objects and match the learned representation to new images. This limitation makes it hard to match general images, where learning is not available. Similarly, approaches which overcome the unreliability of bottom-up segmentations by exploiting top-down knowledge, e.g., [9], [27], or which improve the specificity of the intermediate-level representation by extending it to a hierarchical one, e.g., [53], cannot as yet match general images.

It is generally agreed that the bottom-up computation of intermediate image representations is an important step in organizing the image to facilitate recognition and other processing. However, so far it has been unclear how to fully exploit such representations for recognition. In this paper, we show how to apply such a representation for recognition directly, without requiring extensive learning. We attack the global/local dilemma in shape matching by introducing image segmentation for shape matching. To address the unreliability of segmentations, we propose to average over all possible segmentations weighted by their probabilities conditioned on the images. Our method combines the advantages of global and local approaches: It can match shape globally yet efficiently, and is robust to local-shape variation yet remains sensitive to the detailed boundary shapes. We demonstrate its efficacy in experiments on recognition/localization of an object category, and on tracking.

By connecting segmentation and matching, we can also close the loop and propose new methods for segmentation. We define and compute the *central* segmentation that minimizes the average distance, weighted by probability, to all segmentations of an image. Previous segmentation methods have been plagued by the problem of inconsistent and unpredictable behavior. Since our method finds the

- *H. Wang is with the Penn Image Computing and Science Lab (PICSL), Department of Radiology, University of Pennsylvania, School of Medicine, 3600 Market Street, Suite 320, Philadelphia, PA 19104-2644. E-mail: hongzhiw@mail.med.upenn.edu.*
- *J. Oliensis is with the Department of Computer Science, Stevens Institute of Technology, Castle Point on Hudson, Hoboken, NJ 07030. E-mail: oliensis@cs.stevens.edu.*

Manuscript received 5 Nov. 2008; revised 31 July 2009; accepted 7 Oct. 2009; published online 25 Nov. 2009.
Recommended for acceptance by A. Srivastava, J.N.Damon, I.L. Dryden, and I.H. Jermyn.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMISI-2008-11-0766.
Digital Object Identifier no. 10.1109/TPAMI.2009.199.

0162-8828/10/$26.00 © 2010 IEEE Published by the IEEE Computer Society

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

segmentation of minimum variance, its output can have greater predictability and consistency. We show that this new method gives competitive results on the Berkeley image segmentation database.

As a side product of our segmentation approach, we emphasize the connection between segmentation and edge-preserving smoothing, presenting an algorithm and experiments for the latter. Unlike previous methods (e.g., level set methods or bilateral smoothing), our smoothing exploits the global optimal image structures, not just the local image coherence. It follows the possible segmentations according to their probabilities and is not confined to any single one.

The paper is organized as follows: We first give a brief overview of our approach in Section 2; the technical details of our approach are described in Section 3; in the end, we show our extensive experiments comparing our approach with other related works in image segmentation, image smoothing, object detection, and tracking tasks.

2 Shape Matching: Related Work and Motivations

2.1 Edge-Based Shape Matching

To avoid the potentially exponential cost of matching nondistinctive local fragments, one common solution is to increase the fragments' distinctiveness by upgrading to semilocal descriptors, see, e.g., [5]. Representing shape at a coarse scale or with sparse sampling can further reduce the search space but adds the risk of aliasing, e.g., [14].

A semilocal edge descriptor characterizes the shape geometry over an extended neighborhood. For example, shape context uses a uniform edge histogram in a log-polar space [5]. Such a descriptor is more sensitive to nearby edge points than to those farther away. Spin images use the shape profile over some neighborhood as seen from each edge point [21]. Instead of histograms, geometric blur uses blurred edge maps, with heavier blur for further away edge points to handle larger deformations [6].

These powerful semilocal descriptors make point-to-point matching less ambiguous. However, the computational cost of matching remains too high (usually $O(N^{2-3})$, where N is the number of edge points) for large scale shape matching. Hence, sparse shape sampling is often necessary [5], [7], which decreases the overall shape description accuracy. Similarly, the complexity of hierarchical representations such as in [53] often leads to a requirement for sparse sampling.

To avoid an expensive search over deformations in matching the details of two similar shapes, some approaches first extract a higher level representation of the essential shape structure and match this. For example, one class of approaches transfers shapes into skeletons represented via a graph or tree structure. Shape representations are matched by finding the largest isomorphic subgraph/subtree [54], [42], [44]. Although this method can handle large deformations, it suffers from instability of the derived representation, which complicates matching, and it cannot capture small shape differences well since the skeleton representation compresses the shape variability.

For shapes with small distortion, shape matching is easier and the difficulty of finding high quality point-to-point shape correspondence decreases. For instance, Chamfer matching [3] and Hausdorff-distance matching [20] compute edge correspondence purely based on the euclidean distance between edge points. One drawback of these approaches is that they rely on edge detection, a hard classification decision which is sensitive to noise and illumination changes. Our approach is also based on edges, but relies on the edge probabilities and does not require a hard decision. The recent histogram of gradients (HoG) approach [13] represents shapes as gradient orientation histograms over regions. Matching is done via histogram comparison. This approach implicitly uses the euclidean distance for shape correspondence, but more robustness is achieved via two major improvement: 1) focusing on orientations rather than gradient magnitudes, which gives a degree of photometric invariance; 2) softening the edge correspondence via histogram matching and allowing each edge point to be added in multiple histogram bins. Our approach has similar properties but offers more robustness against photometric variation and failures in edge-to-edge correspondence.

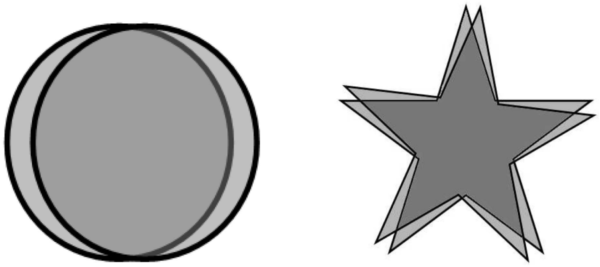


Fig. 1. Even a small shift or rotation can result in a big edge misalignment. Region overlap is less affected by such changes.

2.2 Region-Based Shape Matching

To avoid the local ambiguities of edge matching, one can instead match *regions*. Region matching gives contour matching (for closed contours) with no need for explicit edge correspondence and has a long history [4], [2]. Our approach can be viewed as matching regions.

Fig. 1 illustrates the difference between edge and region matching. As local features, edges can be easily detected and represented locally, but their locality makes comparing them sensitive to spatial shifts and shape variations. Regions are global features, so their matching is more robust to local-shape distortions and occlusions, but they can have complex shapes which may be difficult to extract reliably and hard to represent or match precisely. Since regions have closed boundaries, they don't adapt easily for matching open image curves.

The recent approach of [28], [13], [8] can be thought of as intermediate between region and edge matching since it associates edges with regions by representing shapes as edge orientation histograms over regions. This technique efficiently combines the advantages of local edge representations and global region robustness. A major drawback is that as an information loss method edge histograms sacrifice shape description accuracy.

Probabilistic index maps (PIM) [22] direct each pixel to an image-dependent palette and can efficiently capture the common structures underlying multiple images despite large

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

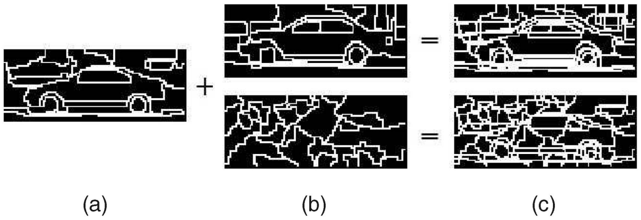


Fig. 2. Segmentation-based shape matching. (a) and (b) A segmentation and two candidate segmentations for matching it. (c) The joint segmentations. Overlapping two similar shapes (top) gives larger regions than overlapping dissimilar shapes (bottom).

appearance changes caused by feature variations, e.g., color or illumination changes. Since training images are required to learn the class structure model, PIM is not suitable for comparing images by their shape structures directly.

2.3 Using Segmentations for Recognition

Our approach implements region-based shape matching and recognition by matching *segmentations*. When applied for recognition, segmentation is most often used to separate the foreground regions or objects of interest from the background (e.g., [19]). Once the potentially interesting regions have been identified, the recognition module focuses on the corresponding subimages and their interrelations. For the related task of classification, a common approach (e.g., [49]) represents each segmented region by a collection of image descriptors and classifies based on the combined vector of descriptors. In addition to these feed forward approaches, some methods use recognition to aid segmentation, either via feedback or by running the two modules jointly [52], [26], [16].

We use segmentation in a different way. Besides separating foreground from background, a segmentation communicates the intrinsic structure of an image. The homogeneous regions of an image can be considered as basic visual units giving a compressed image representation, and the relations between segments, and their locations and shapes, provide the global shape structure of the image (see Fig. 4). The shape of individual segments has been used before, e.g., in [2], but we are the first to use the entire shape encoded in an image segmentation for shape matching.

Utilizing segmentations for region-based shape matching has several advantages: 1) By matching all regions at once, we gain robustness to the grouping failures for any single region; 2) since segmentations are computed using global image information, they can localize the true edges more accurately than local edge-detection/grouping methods; 3) segmentations can reveal global shape structures which are more distinctive than local features. This helps overcome difficulties caused by “hallucinated” boundaries.

A typical segmentation includes both true and hallucinated boundaries. To match segmentations, we need a metric that detects the true matching boundaries while tolerating the hallucinations.

2.4 A Segmentation Similarity Measure

To achieve this, we propose a new similarity measure. It relates to the mutual information (MI) as adapted to segmentations; the basic concept is the *structure entropy* (SE), i.e., the entropy measuring a segmentation’s complexity. A segmentation with many small segments has high SE; large segments give low SE. (Note: We compute SE for individual segmentations, not for the probability distribution over segmentations discussed later.) We define the *structure mutual information* (SMI) between segmentations in terms of their joint segmentation obtained by superimposing the individual ones. When two segmentations have matching boundaries, the joint segmentation has large regions and low SE, so the SMI is high. For nonmatching segmentations, the joint segmentation has smaller regions and higher SE, so the SMI is low. Fig. 2 illustrates the idea. The precise definitions are below.

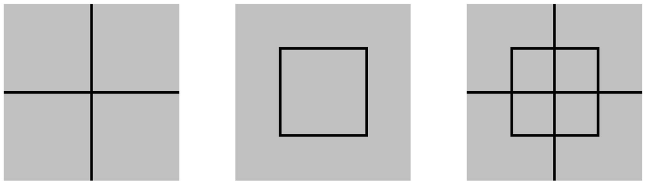


Fig. 3. When edge orientation distribution is used to represent shapes, the left two images are considered to be identical. Our method compares the individual segmentations of these images to their joint segmentation (on the right) and can discriminate between the two shapes efficiently and accurately.

This similarity measure achieves good tolerance against small shape variations. For example, shifting one segmentation a small amount relative to another creates new small regions in the joint segmentation, but large segments continue to have large overlaps, so the joint SE remains low and the SMI remains high. Compared to the less accurate histogram-based shape descriptors, our segmentation-based shape matching can efficiently match the exact boundaries of segmentations (see Fig. 3). HoG [13] can improve the shape representation accuracy by inserting more bins; however, since different bins are evaluated independently, employing fine-scale bins loses integrative power in describing/comparing large scale shapes. In comparison, segmentation boundaries emerge from grouping over large scales yet are good for comparing fine-scale shape differences.

2.5 Segmentation Unreliability

Despite their many good properties as shape representations, see above, segmentations cannot be computed reliably bottom-up from image data. Recognition methods have tried various ways of overcoming this unreliability. Soft integration of multiple regions and the exploiting of statistical interactions between different region types [49], can partly compensate for segmentation unreliability. Alternatives include using multiple segmentations [38] and oversegmentations [35].

For shape matching, oversegmentation is more appropriate since it preserves many of the true shape structures (though these may be hidden within many fake boundaries). Since an oversegmentation includes strong curves, even if they are open, it may be exploited for matching open as well as closed curves. Fig. 4 shows that objects are easy to recognize in oversegmented images, implying that such segmentations contain enough shape information for recognition. For this example, the segmentations vary greatly but a common mid-level car shape structure appears in all car

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

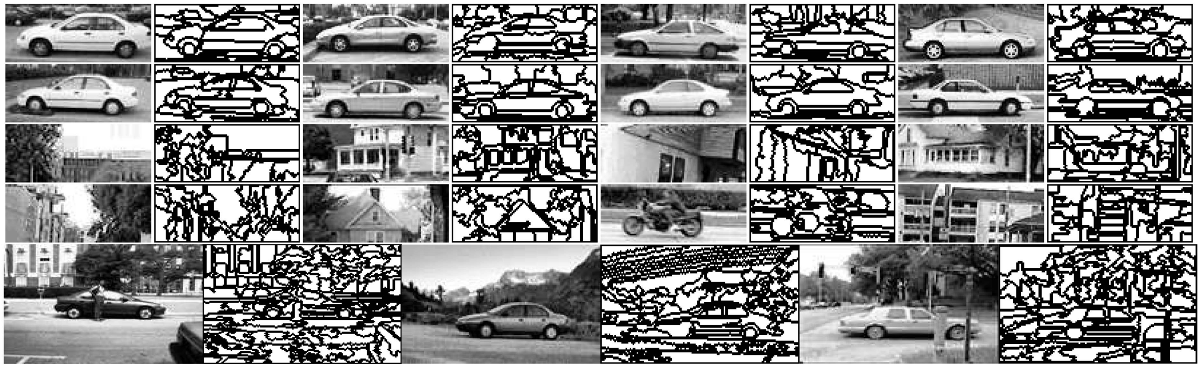


Fig. 4. Segmentation-based shape representations. Despite variations in appearance, the oversegmentations (generated by the mean-shift algorithm EDISON [12] with an average segment size of about 100 pixels) preserve the main image structures.

images, especially the car wheels, window, and side. The fact that people can easily recognize the cars in these oversegmentations despite the low signal/noise ratios for the car edges also shows the distinctiveness of global shapes.

We achieved promising initial recognition results using oversegmentations to represent shape. However, this approach still relies on the quality of the precomputed segmentations. The many fake boundaries in an oversegmentation in effect decrease the accuracy of the shape representation, and this effect becomes stronger for scenes and objects (e.g., human faces) with less distinct boundaries.

2.6 Segmentation Averaging

Our solution to address the unreliability of image segmentation is our main theoretical contribution. Instead of computing actual segmentations, we compute their average similarity, averaging over all possible segmentations for both images weighted by their probabilities conditioned on the image. Indistinct boundaries also contribute to the average, so we get good matching even for unstructured images.

On its face, averaging over all segmentations seems impossible. We cannot do it exactly because we cannot enumerate all possible segmentations even for a small image. We present an accurate approximation giving the average similarity between segmentations in closed form.

Our key realization is that we can compute the averaged structure entropy (and similarly the averaged SMI) separately in terms of each pixel's contribution. We show that a pixel's contribution to the averaged SE is the geometric mean size of the segment containing it. The geometric mean is also hard to compute exactly, but we exploit a standard technique to approximate it in terms of the arithmetic mean. This approximation is a mild one: Roughly, it originates from a Taylor expansion of the geometric mean around the arithmetic mean, and we show both theoretically and experimentally that the higher order terms in this expansion can be neglected. The following sections present the details of our approach.

3 Segmentation-Based Shape Matching

3.1 A Segmentation Similarity Measure

Even though our following discussion can be applied for arbitrary segmentations, since we are interested in exploring the shape structure of image segmentations, we define that two pixels share the same segment label in a segmentation if and only if they are connected by pixels from the same segment. By definition, distant pixels separated by pixels from different partitions do not belong to the same segment.

We start by defining the SE. Given a segmentation s with n segments, we define the structure entropy for s as

$$H(s) = -\sum_{i=1}^{n} p_i \log(p_i), \tag{1}$$

where the sum is over segments and p_i is the area ratio of the ith segment to the whole image (p_i can also be considered as the probability that a random dart thrown at the image will land in the ith segment).

For two segmentations s, s', the joint structure entropy $H(s, s')$ is the structure entropy of the joint, i.e., superimposed, segmentation (it can also be defined as the usual joint entropy in terms of the probabilities for two independent random darts). See Fig. 5a.

Having defined the structure entropy, we define the MI of two segmentations in the standard way (note the semicolon):

$$H(s; s') = H(s) + H(s') - H(s, s'). \tag{2}$$

We call this the *SMI*. The *VoI* is defined by [31]

$$\mathrm{V}(s, s') \equiv 2H(s, s') - H(s) - H(s'), \tag{3}$$

and relates to the MI via $\mathrm{V}(s, s') = H(s, s') - H(s; s')$, see Fig. 5b. The computational cost of SMI and VoI is linear in the image size. Meila [31] showed recently that the VoI gives a distance metric for clusterings. In our context, it gives a distance metric on segmentations.[1]

How do SMI and the SE compare to the standard MI and entropy based on image intensities? Since the intensity entropy has no geometry information, i.e., it does not include spatial interactions between pixels, all permutations of the pixel positions of the image leave the intensity entropy unchanged. In a segmentation, each segment is a spatially connected set of pixels, and only a relatively few pixel permutations, e.g., those giving the same number of regions with the same sizes, have the same SE. Furthermore, perceptually meaningful segmentations usually have large

[1] We derived our measures independently, but later than Meila.

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

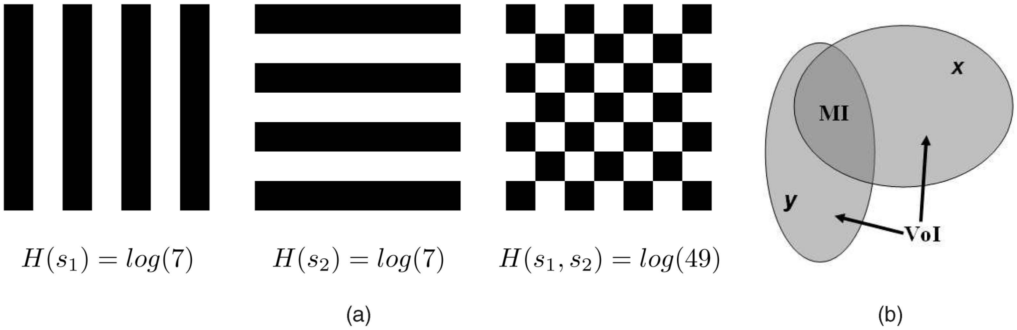


Fig. 5. (a) Two segmentations and their joint segmentation (right). The segmentations have VoI log(49) and MI 0. (b) The relation between the MI and the variation of information (VoI).

size segments, with small SEs. High SEs indicate many small segments, which usually have trivial, meaningless structures. In comparison, high-intensity entropy does not necessarily indicate meaningless images. For example, a natural shading may cause the brightness to change uniformly and gradually from brightest on one side to darkest on the other. Such an image has the highest intensity entropy. In short, segmentation and SE enforce the connectivity and proximity Gestalt rules, which characterizes the image geometric structure more accurately and conveys more meaningful structural information than intensity entropy.

Russakof et al. [36] include spatial information for intensity MI by considering regions instead of single pixels in computing entropy. To address the difficulty in probability density estimation in high-dimensional space, this work assumes that the density function is Gaussian. In comparison, we include spatial information in a more general way, without such a strong assumption.

Both the VoI and SMI are *biased* with respect to the complexity of the compared segmentations. Two different segmentations may have high SMI (high similarity) just because they both have high structure complexity, or low VoI distance (again, high similarity) because they both have low complexity (since $V(s, s') \leq H(s, s') \leq H(s) + H(s')$). For instance, Fig. 6 shows two square shapes. The spatial shift between them causes a VoI distance of 0.71, which is higher than the VoI distance, 0.56, between either of them and a null shape (the image is a single segment). However, measured by SMI, they are more similar to each other than to the null shape.

In applying the metric VoI for matching and recognition, we compensate for its bias based on the following heuristic argument: A complex shape is distinctive, and if we find a reasonable match, it is likely to come from the same object, whereas different simple objects are more likely to have similar shapes, just by virtue of their simplicity, so we should require close similarity before we declare two simple shapes as the same object. We impose this preference by normalizing VoI as follows:

$$\begin{aligned}\Delta(s, s') &= \mathrm{V}(s, s')/H(s, s') \\ &= 1 - H(s; s')/H(s, s') = \frac{1}{1 + H(s; s')/\mathrm{V}(s, s')}.\end{aligned} \tag{4}$$

The measure Δ also has the advantage of taking values just in the interval $[0, 1]$, with $\Delta = 0$ for a perfect match and $\Delta = 1$ for a perfect mismatch. This standardizes the measure across different shapes, making it easier to interpret. Finally, note that Δ is still a distance metric for segmentations (proof in Appendix A). After normalization, the two segmentations in Fig. 6 have a distance 0.77 and they both have the largest distance, 1, to the null shape.

To test the performance of the normalized VoI, we use it to measure the consistency of segmentations labeled by different human subjects on the Berkeley segmentation database. Fig. 7 gives results for subjects 1105 and 1109. The human results show good consistency by our measure, but still there are big inconsistencies. These come mainly from a difference in the amount of detail labeled. Our measure does capture the similarity of the details labeled by both subjects, but details labeled by just one reduce the similarity score. Recall that identical segmentations give a perfect score of 0 and unrelated images typically score near 1. For more intuition, Table 1 gives the similarities between

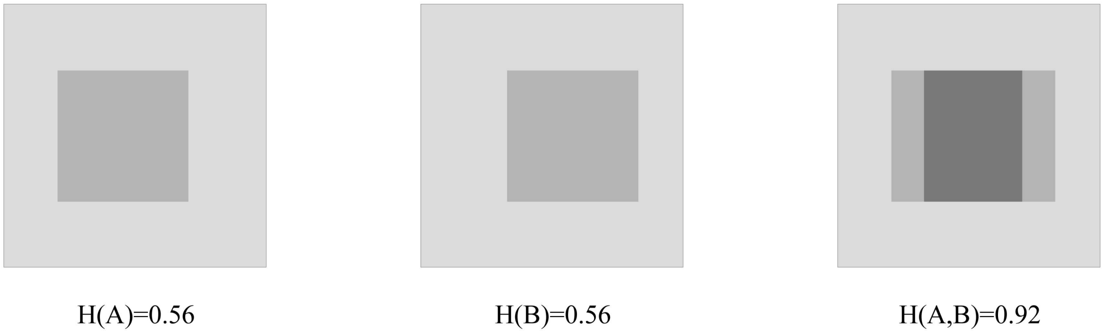


Fig. 6. Two segmentations, A and B, with square shapes (left two) and their joint segmentation (right). $\mathrm{V}(A, B) = 0.72$. The VoI distance between A and the null shape, O, is $\mathrm{V}(A, O) = 2H(A, O) - H(A) - H(O) = H(A) = 0.56$.

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

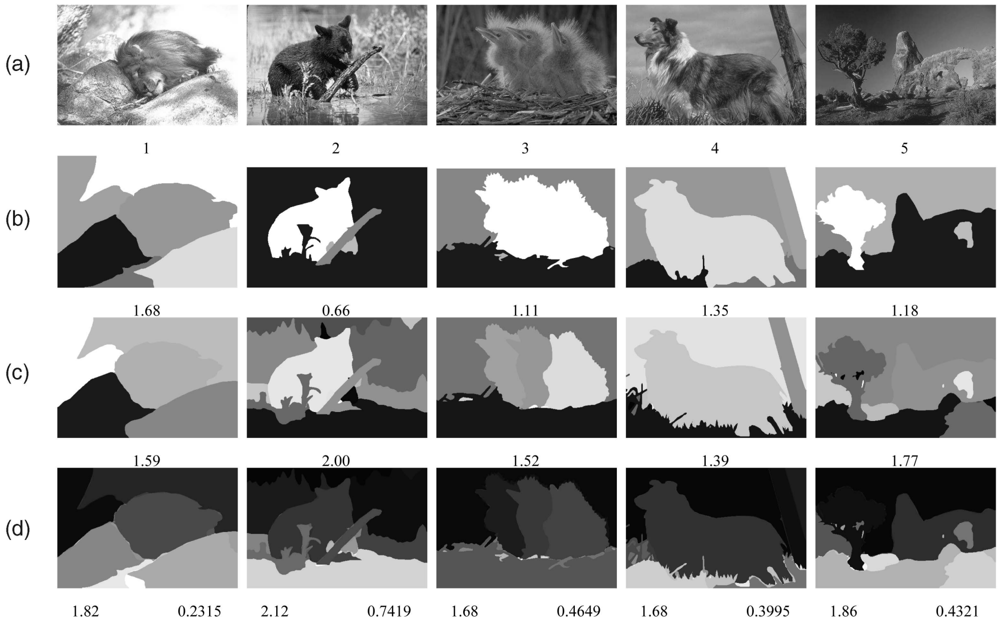


Fig. 7. Segmentation consistency by human subjects. (a) Original images; (b) and (c) segmentations by subjects 1105 and 1109, respectively, with structure entropy; (d) joint structures with the structure entropy and similarity values.

segmentations of the five different images as labeled by the two subjects. The unrelated segmentations clearly are less similar than those for the same image (on the diagonal).

To verify our previous claim that segmentation-based shape matching is robust against spatial shifts, Fig. 8 gives the average VoI between a segmentation (from subject 1109 in Fig. 7) and its shifted version over a range of shifts along the x-axis. For comparison, we also give the average distance using intensity MI. As we can see, MI on intensity performs poorly, with the most similarity loss on the first pixel shifted, while our segmentation-based approach shows good robustness to position shifts. The sensitivity to small dislocation makes intensity MI a better candidate for image registration [47], while its insensitivity makes SMI a better candidate for recognition where exactly matched shapes are rare.

3.2 The Averaged Structure Entropy

As described in Section 2, to address the unreliability of segmentations, we match images by computing the average similarity of their segmentations. To this end, we need the average of the structure entropy for an image over all possible segmentations. This section describes our main theoretical contribution: an approximation to this average.

TABLE 1
Similarity Confusion Matrix (Normalized VoI) between Segmentations of the Five Images in Fig. 7 as Labeled by Subjects 1105 (Rows) and 1109 (Columns)

	1	2	3	4	5
1	0.23	0.73	0.72	0.84	0.72
2	0.92	0.74	0.90	0.94	0.94
3	0.72	0.80	0.46	0.86	0.79
4	0.85	0.83	0.89	0.40	0.84
5	0.80	0.78	0.84	0.86	0.43

Let S denote the set of all possible segmentations of an image F. We define the *averaged structure entropy* ASE of F by

$$H_\omega(F) = \sum_{s\in S} p(s \mid F)H(s), \tag{5}$$

where $p(s \mid F)$ is the conditional probability of a candidate segmentation s given F.

Our approximation starts from the observation that, for a given segmentation s, its SE can be rewritten as the sum over each pixel's contribution. The contribution of the mth pixel is:

$$H^{(m)}(s) = -A_F^{-1}\log\left(A_F^{-1}A\left(s^{(m)}\right)\right) \sim \log\left(A\left(s^{(m)}\right)\right), \tag{6}$$

where $s^{(m)}$ is the segment containing pixel m, $A(\cdot)$ gives its area, and $A_F \equiv A(F)$ is the total area of F. The SE is the sum of $H^{(m)}(s)$ over all pixels.

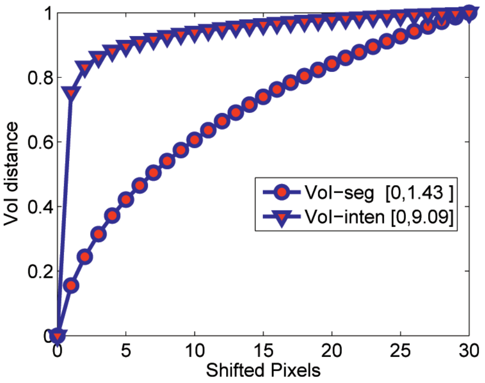


Fig. 8. Robustness test regarding position shifts using VoI on segmentations and intensity images. For better comparison, we display the VoI distances in the same range. The actual value ranges are given in the figure.

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

Similarly, the ASE $H_\omega(F)$ can be rewritten as the sum over each pixel's contribution as well and the contribution of the $m\text{th}$ pixel is:

$$H_\omega^{(m)} = \sum_{s\in S} p(s \mid F) H^{(m)}(s) \sim \log\left[\prod_{s\in S} A(s^{(m)})^{p(s|F)}\right]. \tag{7}$$

Our key insight is: We can evaluate the ASE *without enumerating all segmentations* if, for each pixel in the image, we can compute the geometric mean segment size $G \equiv \prod_{s\in S} A(s^{(m)})^{p(s|F)}$.

The geometric mean is hard to compute. A common approximation [51] is $G \approx \mu - \sigma^2/2\mu$, where μ is the arithmetic mean and σ^2 is the variance. Then,

$$G \approx E\{A(s^{(m)})] - \mathrm{Var}\{A(s^{(m)})\}/(2E\{A(s^{(m)})\}), \tag{8}$$

where E and Var denote the expectation and variance.

The arithmetic mean and variance of segment size can be conveniently estimated in terms of the *affinity matrix* M_F, defined by

$$M_F(m,n) = \sum_{s\in S} p(s \mid F)\chi_{m,n}(s), \tag{9}$$

where $\chi(s)$ is the *segmentation affinity matrix* for the segmentation s, with entries 1 (the pixels belong to the same segment) or 0. We can also consider $\chi_{m,n}(s)$ as an indicator variable representing the event that the given pixels belong to the same segment. For two separate images, it is easy to verify that the *joint* segmentation affinity matrix of their respective segmentations s_1 and s_2 is $\chi_{m,n}(s_1, s_2) = \chi_{m,n}(s_1)\chi_{m,n}(s_2)$. For an image with N pixels, $M_F \in \Re^{N\times N}$, and each entry measures the probability that the two pixels lie in the same segment.

Below, for compactness, we suppress the dependence of $\chi_{m,n}(s)$ on s. Rewriting (9) as $E\{\chi_{m,n}\} = M_F(m,n)$, we derive the mean segment size for the $m\text{th}$ pixel as

$$\mathrm{E}\{A(s^{(m)})\} = E\left\{\sum_n \chi_{m,n}\right\} = \sum_n M_F(m,n). \tag{10}$$

To apply in practice, we estimate M_F from local image properties, e.g.,

$$M_F(m,n) \approx \begin{cases} exp\left(\frac{-|F_m - F_n|}{2\sigma}\right), & \text{if } d(m,n) \le D; \\ 0, & \text{if } d(m,n) > D, \end{cases} \tag{11}$$

where F_m is the intensity, $d(m,n)$ the distance between pixels m, n, and σ the standard deviation of intensity within a segment. Equation (11) embodies the principle that nearby pixels with similar intensity are more likely to group than distant pixels with different intensities. Other cues, e.g., texture or the presence of an intervening edge, could be used as well.

For the variance appearing in (8), we have the upper bound

$$\mathrm{Var}\{A(s^{(m)})\} = \sum_{k,l} cov(\chi_{m,k}, \chi_{m,l}) \tag{12}$$

$$= \sum_{k,l} E[\chi_{m,k}\chi_{m,l}] - E[\chi_{m,k}]E[\chi_{m,l}] \tag{13}$$

$$\le \sum_{k,l} \min(E[\chi_{m,k}], E[\chi_{m,l}]) - M_F(m,k)M_F(m,l) \tag{14}$$

$$= \sum_{k,l} \min(M_F(m,k), M_F(m,l)) \\ (1 - \max(M_F(m,k), M_F(m,l))), \tag{15}$$

where the inequality follows since $\chi_{m,k}\chi_{m,l} \le \chi_{m,k}$ and $\chi_{m,k}\chi_{m,l} \le \chi_{m,l}$. We apply this bound to simplify our expression (8) for the geometric mean, arguing that the first term in (8) dominates the second and by itself gives a good approximation to the geometric mean.

To achieve this, we estimate the bound by treating the pairwise probabilities (that two pixels belong to the same segment) as independent. Many algorithms rely on a similar assumption, e.g., NCuts [40] neglects higher order interactions between $\ge$3 pixels, determining the likelihood of a segmentation from the pairwise affinities only. Independence is usually a strong assumption, but not critical for us since we use it in a weak way and can validate the approximation based on it experimentally, see below. Under the independence assumption, we have

$$\mathrm{Var}\{A(s^{(m)})\}_I = \sum_k cov(\chi_{m,k}, \chi_{m,k}) = \sum_k E[\chi_{m,k}^2] - E[\chi_{m,k}]^2 \\ = \sum_k M_F(m,k)(1 - M_F(m,k)). \tag{16}$$

Combining this with (10), we derive the upper bound

$$\frac{\mathrm{Var}\{A(s^{(m)})\}_I}{2E\{A(s^{(m)})\}} < 0.5$$

on the second term in our expansion (8) for the geometric mean. If $\mathrm{Var}\{A(s^{(m)})\}_I$ gives a "good enough" approximation to the variance upper bound in (15), as our experiments below verify, the true value

$$\frac{\mathrm{Var}\{A(s^{(m)})\}}{2E\{A(s^{(m)})\}}$$

will also be small (and it could be much smaller than 0.5 since this is an upper bound on the value computed from $\mathrm{Var}\{A(s^{(m)})\}_I$ and (15) is an upper bound on the true variance). Since meaningful segments are sizable, we expect $E\{A(s^{(m)})\} = \mu \gg 0.5$, implying that the first term dominates (8) and the arithmetic mean approximates the geometric mean well. We use this approximation in all our experiments, taking $H_\omega(F) \sim \sum_m \log(\sum_n M_F(m,n))$.

Note that our main approximation above is our expansion (8) of the geometric mean in terms of the arithmetic mean. We use independence only indirectly, to support our neglect of the second term of (8), and our experiments below validate this approximation.

3.2.1 Validation of Pairwise-Independence and the Arithmetic Mean Approximation

We approximated the geometric mean by the arithmetic one assuming pairwise independence. For us, this assumption is especially appropriate, since we only keep the affinities of nearby pixels (only these can be reliably estimated) which are mostly near 1 with small covariances, see (15).

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

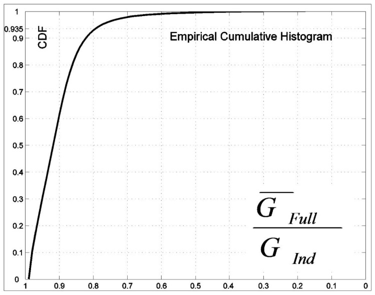


Fig. 9. Empirical distribution of $\frac{\bar{G}_{Full}}{G_{Ind}}$, verifying that the independence assumption as used for (16) doesn't compromise our conclusion that the geometric mean of segment size is well approximated by the arithmetic mean.

To verify our approximation, Fig. 9 shows the empirical distribution of $\frac{\bar{G}_{Full}}{G_{Ind}}$ over real images, where $\bar{G}_{Full}$ is the lower bound derived from (15) on our expression (8) for the geometric mean (no independence assumption), and G_{Ind} is the value of (8) computed assuming independence. As the plot shows, the ratio is above 0.8 for over 93.5 percent of the pixels. It is easy to see that G_{Ind} gives an upper bound on (8), and $\bar{G}_{Full}$ gives a lower bound, so the plot implies that they both approximate (8) well. Hence, from our argument above that the arithmetic mean approximates G_{Ind}, we conclude that it also approximates (8) well.

The images used were the Berkeley segmentation training data [29]. For faster computation, we resized the 200 images into 80×120 using the Matlab bicubic function. For any image, each pixel gives a sample of the ratio. The Gaussian function (11) with deviation $\sigma = 10$ (intensity range $[0, 255]$) and $D = 5$ is used to compute affinity matrix.

3.2.2 A Toy Example

For intuition, Fig. 10 gives a simple image with 4 pixels, plus all 15 possible segmentations and their SEs. Assuming each segmentation is equally likely, with probability $1/15$, we can compute the ASE for this example, which is:

$$H_\omega(F) = \frac{1}{15}\log(4) + \frac{6}{15}\left[-\frac{1}{2}\log\left(\frac{1}{2}\right) - \frac{1}{2}\log\left(\frac{1}{4}\right)\right] + \frac{3}{15}\log(2) + \frac{4}{15}\left[-\frac{3}{4}\log\left(\frac{3}{4}\right) - \frac{1}{4}\log\left(\frac{1}{4}\right)\right] = 0.7969.$$

To apply our approximation, we construct the image affinity matrix by averaging across all segmentations, i.e., (9), and obtain:

$$M = \begin{pmatrix} 1 & 1/3 & 1/3 & 1/3 \\ 1/3 & 1 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1 & 1/3 \\ 1/3 & 1/3 & 1/3 & 1 \end{pmatrix}.$$

The triplet affinity that any 3 pixels lie in one segment can be computed similarly, which is a constant $2/15$. The arithmetic mean of segment size containing each pixel is $E\{A\} = 1 + 3(1/3) = 2$. The variance of the segment size is: $\mathrm{Var}\{A\} = 3(1/3)(1 - 1/3) + 6(2/15 - 1/9) = 0.80$. The approximated ASE using the full approximation (8) is 0.7985 with error 0.0016. With the independence assumption, our approximation to H_ω using (8) and (16) is 0.7802 with error 0.017. Note that the theoretical upper bound for the pairwise affinity covariance is $2/9 \approx 0.22$, which is much higher than the real value $2/15 - 1/9 \approx 0.02$. For this toy example, our approximation is quite accurate.

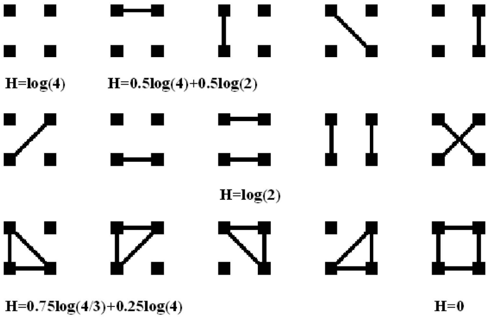


Fig. 10. The 15 segmentations and their structure entropies for a 4-pixel image. Connected pixels are in the same segment.

3.2.3 A Modification

Our approximation to ASE is based on the estimated

$$p^{(m)} \equiv \frac{E\{A(s^{(m)})\}}{A_F}$$

for each pixel m. In practice, the pairwise affinity estimation raises a potential issue. Due to the difficulty of perceptual organization, estimates of the pairwise affinities are reliable only over a small region, especially when the underlying assumption used for computing pairwise affinities does not fit the image well. For instance, the homogeneous assumption underlying (11) is often invalid for large regions. Hence, D in (11) should be small. When D is small, the estimated $A(s^{(m)}) = \sum M_F(m,n)$ tends to underestimate the expected segment size, resulting in inaccurate $p^{(m)}$. Since each pairwise affinity can be considered as a random sample of $p^{(m)}$, when the pairwise affinities are not estimated for all pixel pairs, we estimate $p^{(m)}$ as the mean of all estimated pairwise affinities, which redefines the ASE as:

$$H_\omega = -\frac{1}{A_F}\sum_m \log\left(\frac{E\{A(s^{(m)})\}}{A_m}\right), \tag{17}$$

where A_m is the neighborhood size for the mth pixel. Note that the normalization by the neighborhood area gives an approximation to the structure entropy when not every pairwise affinities can be estimated. This normalization does not change properties of entropies, i.e., H_ω is nonnegative and proportional to the structure complexity.

3.3 Shape Similarity from Averaging Segmentations

We compare the shapes in two images by computing the average distance between the image segmentations. Recall the definition (3) of VoI, which gives a metric on segmentations. Its value, averaged over all possible segmentations for two images F_1 and F_2, is $V_\omega(F_1, F_2) \equiv 2H_\omega(F_1, F_2) - H_\omega(F_1) - H_\omega(F_2)$, where the $H_\omega(F_a)$ are the ASE for the two images, and

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

$$\begin{aligned} H_\omega(F_1,F_2) &= \sum_{s_1}\sum_{s_2} p(s_1,s_2 \mid F_1,F_2)H(s_1,s_2) \\ &= \sum_{s_1}\sum_{s_2} p(s_1 \mid F_1)p(s_2 \mid F_2)H(s_1,s_2) \end{aligned} \tag{18}$$

is the ASE of the joint segmentations for the two images. As before, we approximate $H_\omega(F_1,F_2) \approx \sum_m \log(\sum_n M_{F_1F_2}(m,n))$; because the two images and their segmentations are independent, the joint affinity matrix is easily seen to be:

$$M_{F_1F_2}(m,n) = M_{F_1}(m,n)M_{F_2}(m,n). \tag{19}$$

Note that V_ω is not a distance metric any more. It is nonnegative, symmetric, and obeys the triangle inequality (proof in Appendix B). However, $V_\omega(F,F)=0$ only when F is a segmentation. For real images with multiple plausible segmentations, $V_\omega(F,F)>0$. We can consider the lack of "indiscernibility of identicals" as due to the uncertainty in visual perception. Ambiguity is intrinsic to perception, with many factors contributing to it. For instance, because of noise, discretization, lighting, and shading effects, it is possible that two images with the same intensity pattern represent different scenes. Accommodating such uncertainty is crucial for modeling intelligent vision systems. In our context, V_ω captures the uncertainty reflected in the fact that real images have many plausible segmentations, and $V_\omega(F,F)$ indicates the ambiguity level of image F. When $V_\omega(F,F)=0$, F only has one plausible segmentation with no ambiguity. In the other extreme, an image F with all segmentations equally possible, we get the most ambiguity and the largest value of $V_\omega(F,F)$. As the ambiguity level increases in an image, it has more overlaps with other images and the ability to discriminate it from others decreases. Hence, our method works the best for images with fewer prominent plausible segmentations.

Because of its denominator, it is hard to approximate the average similarity of segmentations using the *normalized* VoI distance of (4). Thus, we compensate for the bias of VoI in a different way: Instead of averaging the normalized distance, we normalize the averaged distance, using

$$\Delta(F_1,F_2) \equiv \frac{V_\omega(F_1,F_2)}{H_\omega(F_1,F_2)} \approx \mathbf{E}\left\{\frac{V(s1,s2 \mid F_1,F_2)}{H(s1,s2 \mid F_1,F_2)}\right\}$$

for image comparisons. Finally, we derive

$$\begin{aligned} \Delta(F_1,F_2) &\approx 2 - \frac{\sum_m \log\left(\sum_{nn'} M_{F_1}(m,n)M_{F_1}(m,n')\right)}{\sum_m \log\left(\sum_n M_{F_1}(m,n)M_{F_2}(m,n)\right)} \\ &= 2 - \frac{\sum_m \log\left(\bar{A}_{1m}\bar{A}_{2m}\right)}{\sum_m \log\left(\bar{A}_{Jm}\right)}, \end{aligned} \tag{20}$$

where $\bar{A}_{am}$ gives the average size of the containing segment for the individual or joint segmentations. Roughly, Δ measures the statistical independence of the segment sizes in the two images. Like V_ω, our normalized $\Delta(F_1,F_2)$ is not a distance metric but it is non-negative, symmetric, and obeys the triangle inequality (see Appendix C for proof). Section 5 applies this Δ in tracking and detection experiments.

Image matching by our segmentation averaging has the same robustness to noise and spatial shifts as using individual segmentations for shape matching. However, we no longer need a previously computed segmentation, and our results are affected much less by the difficulty of finding a correct segmentation. A drawback is that our computational cost becomes $O(dN)$, where d is the neighborhood size in computing pixel-wise affinities.

3.3.1 Comparison to Other Measures

Unnikrishnan et al. [46] has applied the Probability Rand (PR) index to evaluate the consistency of a given segmentation with a set of human segmentations. In our context, we can consider it an alternative similarity metric based on affinity matrices:

$$\begin{aligned} PR(F_1,F_2) &= \frac{1}{dN}\sum_{m,n}[M_{F_1}(m,n)M_{F_2}(m,n) \\ &\quad + (1-M_{F_1}(m,n))(1-M_{F_2}(m,n))] \\ &\propto \sum_m \left[2\sum_n M_{F_1F_2}(m,n) \right. \\ &\quad \left. - \sum_n M_{F_1}(m,n) - \sum_n M_{F_2}(m,n)\right]. \end{aligned} \tag{21}$$

The similarity template (ST) [43] defines another similarity measure based on affinity matrices:

$$ST(F_1,F_2) = \sum_m \frac{\sum_n M_{F_1F_2}(m,n)}{\sum_n M_{F_1}(m,n)\sum_n M_{F_2}(m,n)}. \tag{22}$$

Our averaged MI replaces the sum in ST by a product:

$$H_\omega(F_1;F_2) \propto \prod_m \frac{\sum_n M_{F_1F_2}(m,n)}{\sum_n M_{F_1}(m,n)\sum_n M_{F_2}(m,n)}. \tag{23}$$

PR evaluates individual affinity independently, totally ignoring the spatial interaction between pixels. ST and our method include the spatial interaction. Recall that $\sum_n M(m,n)$ is proportional to the expected segment size for the $m\text{th}$ pixel. Hence, our metric evaluates the statistical independence of segments. The difference from ST is that we combine the independence measures for different pixels by taking a product, which has a clear entropy meaning, while ST uses summation, whose statistical meaning is not clear. For these reasons, we expect our approach to perform the best and ST better than PR. Our detection experiment confirms this claim.

4 Application to Segmentation and Smoothing

As researchers increasingly focus on the recognition problem, they have also devoted more attention to *mid-level vision*, especially *segmentation* [12], [15], [39], [40]. In this section, we describe how image segmentation can benefit from our approach.

Several recent segmentation advances [39], [40] use new methods to minimize standard cost functions that embody Gestalt grouping criteria such as similarity and proximity. Interpreting the costs as $-\log(\text{probability})$, one can view these approaches as computing the Maximum a Posteriori (MAP) segmentation. Though it has no explicit cost function, one can also categorize mean shift [12] as a MAP method.

Our work starts from the observation that MAP is not the best estimator for segmentations. The "best" segmentation is often ambiguous and difficult to find, which suggests that the probability distribution for segmentations is likely to

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

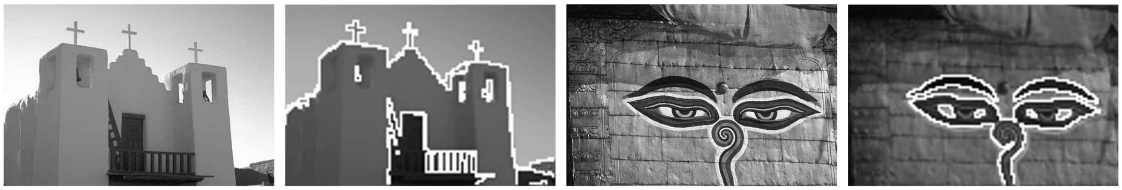


Fig. 11. Segmentation results for naive greedy merging. $D = 5$, image size 80×120.

have a broad rather than narrow peak around its maximum. The fact that people, and algorithms based on similar models, can produce very different segmentations (see Fig. 15) is also suggestive of this. The imbalance between small and large segments suggests that the probability is highly asymmetric. For random variables with such distributions, the distribution *mean* is the estimator with least variance and usually superior to the MAP. Another way to put it is that the huge difficulty and ambiguity of *bottom-up* perceptual organization makes the behavior of existing segmentation algorithms sensitive to the input images and parameter settings. Since the quality of results may vary greatly depending on the image and parameters, obtaining good results on different images depends heavily on trying out different parameter settings. For a given algorithm, the difficulty in choosing the optimal parameter values and the algorithm's sensitivity to these values determines how well it can do. Computing the mean estimator for image segmentation gives more consistent and more predictable results, which reduces the dependence on optimal parameter selection. To estimate the mean, one must average over all segmentations weighted by their probabilities. Our technique directly supports this estimation.

The most related work [18] also segments by averaging over segmentations. Differences include: Gdalyahu et al. [18] average using *sampling*, which is slow, and it computes the mean affinity matrix, not the segmentation directly. Segmentation is obtained by hard thresholding local affinities. Our approach directly estimates the mean segmentation from the affinities and can enforce global consistency among the local affinities. Moreover, our result is a closed form and applies more generally, e.g., to matching, while Gdalyahu et al. [18] focus just on segmentation.

In the context of utilizing shape priors to guide image segmentations, Joshi et al. [23] and Charpiat et al. [11] compute the mean shape for bounding contours, i.e., silhouettes, while our method can be applied to any segmentation.

4.1 The Central Segmentation

Since the mean has least variance, we can compute it for an r.v. x as $\bar{x} = \arg\min_x \sum_y p(y)|y-x|^2$. Recall that $V(s, s')$ gives a metric for segmentations. Given an image F, we define its *central segmentation*

$$\hat{s} \equiv \arg\min_s \sum_{s' \in S} p(s' \mid F) \mathrm{V}(s', s) \equiv \arg\min_s V_\omega(s, F). \tag{24}$$

$\hat{s}$ is "central" in that it minimizes the average distance V to all segmentations of F.[2] It is the mean segmentation with respect to the distance metric $\sqrt{V}$. We choose $\hat{s}$ partly for convenience since we already approximated V_ω; also, we expect *better* segmentations using $\sqrt{V}$ and $\hat{s}$ than for the metric V.

Our reasons for this expectation are as follows: A typical image has many qualitatively different yet plausible segmentations, implying $p(s \mid F)$ has many large peaks. Averaging over all s combines qualitatively different segmentations, whereas we want the center of the dominant peak. Using the metric $\sqrt{V}$ for the average gives a robust estimate with more resistance to outlier segmentations.

For the same reasons as in the previous section, we normalize the averaged distance, redefining

$$\hat{s} \equiv \arg\min_s \frac{V_\omega(s, F)}{H_\omega(s, F)} \equiv \arg\min_s \Delta(s, F),$$

where $H_\omega(s, F) \equiv \sum_{s' \in S} p(s' \mid F) H(s, s')$.

4.2 Segmentation Algorithms

Finding the central segmentation is a combinatorial optimization problem. For intuition on the difficulty of this problem, we first tried to optimize our cost via a greedy merging algorithm. We start with each pixel as a segment, then merge neighbor segments if and only if this decreases $\Delta(s, F)$. Here, and in our segmentation and smoothing experiments, we used the simple affinity matrix M_F of (11) to compute V_ω, H_ω, and Δ. Because M_F is nonzero just over small neighborhoods, the greedy approach can compute the merged segmentation with linear cost $O(Nd)$, where d is the neighborhood size determined by D in (11). Fig. 11 shows this naive method can give excellent results, indicating the robustness of our criterion $\Delta(s, F)$.

For a more complete search, since our cost function combines local interactions between pixels, it is difficult to apply advanced methods such as graph cut [10], which requires isolated, regularized pairwise energies. Instead, we guide our search using an approximate version of steepest descent. We consider the segment labels as continuous variables and iterate using approximate variational optimization to propose label changes that are likely to reduce the cost. Also, to avoid local minima, we use a type of regularization to focus the search around the global optimum's likely locations. Note that we always evaluate a label configuration by computing its exact cost $\Delta(s, F)$; our variational method is just a way of finding good incremental label changes, i.e., a strategy for searching over label configurations. The strategy is heuristic but in practice works well. The greedy method's good results suggest that our optimization is not difficult and should not depend sensitively on the details of our search algorithm.

Recall that the structure entropy of a segmentation is

$$H(s) = -\sum_m \log \frac{\sum_n I(s^m = s^n)}{A_m},$$

[2] To compute the "central" segmentation for several plausible segmentations labeled by algorithms and humans, instead of averaging over all possible segmentations one can average over the several most plausible segmentations as well.

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

where I is an indicator function.[3] To apply variational optimization, we represent a segmentation by *real-valued* labels s_m at each pixel m, and write the structure entropy and its average as

$$H(s) = -\sum_m \log \frac{\sum_n I(\mathrm{r}(s_n) = \mathrm{r}(s_m))}{A_m}, \tag{25}$$

$$H_\omega(s, F) = -\sum_m \log \frac{\sum_n I(\mathrm{r}(s_n) = \mathrm{r}(s_m)) M_F(m,n)}{A_m}, \tag{26}$$

where r is the round function. Since in this paper we segment based just on the intensity, we will initialize our search with the labels s_m equal to the intensities of the original image. This choice gives a meaning during the search to the distance $|s_m - s_n|$: Pixels with well-separated labels are less likely to end up assigned to the same segment.

To compute an approximate, well-behaved "steepest descent direction" for our cost Δ, we seek a smooth approximation to the derivatives of the numerators in (25) and (26). For the exact expressions with discrete labels, incrementing a label (i.e., increasing it by 1) changes an indicator function only if the other label appearing in the indicator is close to the incremented label (i.e., at most 1 away from it). If the other label is far from the incremented one, there is no change. For the continuous representation, we implement a softened version of this, replacing the indicator $I(\mathrm{r}(s_n) = \mathrm{r}(s_m))$ by a function $G(s_n - s_m)$ which decreases with $|s_n - s_m|$. Another motivation is that the most important changes to explore during search are those where one label moves away from another that is close in value; labels that are far in value from the given label most likely comes from a different segment (because of our initialization), and they should have little influence on the best assignment of the label among the relevant segments with close labels. For faster convergence, the influence of one label on another shouldn't decay too fast with their separation in value. Based on these heuristics, we adopt a label "influence function" that decays at large value separation as $G(s_m - s_n) \sim \exp(-|s_m - s_n|^{0.5})$. This form gives singular behavior at small label separations and must be cut off for stability. Replacing $I(\mathrm{r}(s_n) = \mathrm{r}(s_m))$ by the cutoff $G(s_m, s_n)$ and computing the numerator derivatives, we get

$$\frac{\partial H(s)}{\partial s_m} \approx \sum_{n \in \mathcal{N}(m)} \frac{\mathbf{sign}(s_m - s_n)\max(1, |s_m - s_n|)^{-0.5}}{\sum_{k \in \mathcal{N}(n)} I(\mathrm{r}(s_k) = \mathrm{r}(s_n))}, \tag{27}$$

$$\frac{\partial H_\omega(s, F)}{\partial s_m} \approx \sum_{n \in \mathcal{N}(m)} \frac{\mathbf{sign}(s_m - s_n)\max(1, |s_m - s_n|)^{-0.5} M_F(m,n)}{\sum_{k \in \mathcal{N}(n)} I(\mathrm{r}(s_k) = \mathrm{r}(s_n)) M_F(k,n)}. \tag{28}$$

In these equations, we have made the additional simplification of eliminating a factor G; this increases the range of influence slightly and seems to give faster convergence. The cutoff $\max(1, |s_m - s_n|)$ adds stability; also, as is true for the exact discrete cost, it helps ensure that label values from the same segment all produce the same value 1 for the derivative, at least for the most relevant segments (with label value close to the other label in the indicator).

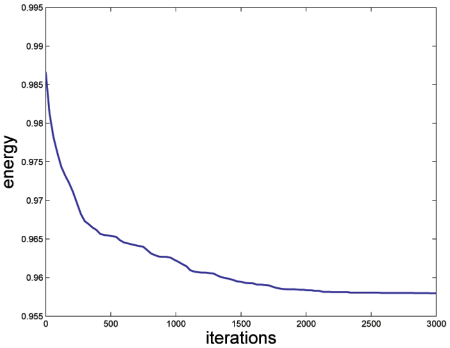


Fig. 12. Illustration of the convergence on a test image with size 160×240.

For even faster convergence, we add a "force" term $0.1 \frac{\partial H_\omega(s,F)}{\partial s_m}$ to the gradient. This acts as a regularization that focuses the search on simpler segmentations, helping overcome local minima. At the end of our search, we always output the segmentation with minimal $\Delta(s, F)$; the algorithm above is just a search strategy. The C implementation of our search algorithm[4] usually takes a few minutes to converge to a local optima on a 3.0 GHz AMD for images with size 160×240 (code available on our web page). Fig. 12 illustrates the convergence of our segmentation algorithm.

4.3 Smoothing

Segmentation relates closely to edge-preserving smoothing; in fact, one can consider it as a piecewise constant smoother. We implement edge-preserving smoothing by finding the "most similar" image to the original image F according to our criterion Δ. The algorithm is steepest descent, similar to the segmentation procedure (but without the derivative approximation or extra force), except we optimize Δ over images instead of segmentations.

The most similar image F_{sim} does *not* equal the original. Instead, one can show that it has a segmentation probability distribution which clusters around $\hat{s}$, the central segmentation of F. As a result, F_{sim} agrees with the boundaries of F and varies smoothly over its nonboundary regions (to discourage unlikely segmentations). Unlike previous methods based on local computations [34], [45], our approach smoothes according to the image's *global* optimal structures. By averaging over probabilities, it can adjust the smoothing according to boundary strength and smooth *across boundaries*, not just within segments. Fig. 13 shows some smoothing results. Our method gives appealing smoothings which preserve contours, with global salient structures enhanced.

[3] To match a segmentation with a real image, we use the same neighborhood to compute pair-wise affinities for both the segmentation and the image.

[4] The algorithm used differs slightly from the one presented here; the denominators have $I(|s_m - s_n| \leq 1)$ instead of $I(r(s_m) = r(s_n))$.

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

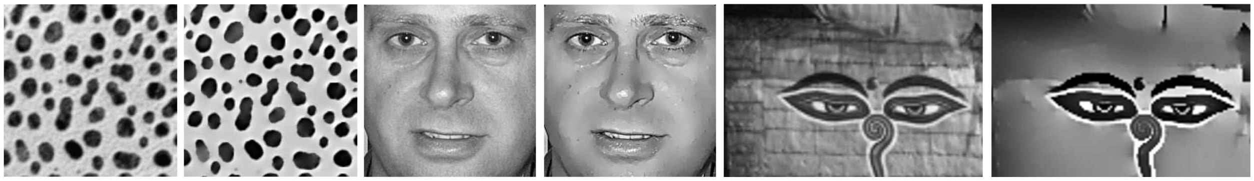


Fig. 13. First, third, and fifth columns: the original images; second, fourth, and sixth columns: smoothing results by our method after 30 iterations. $D = 2$. The σ used are 60, 20, and 60, respectively.

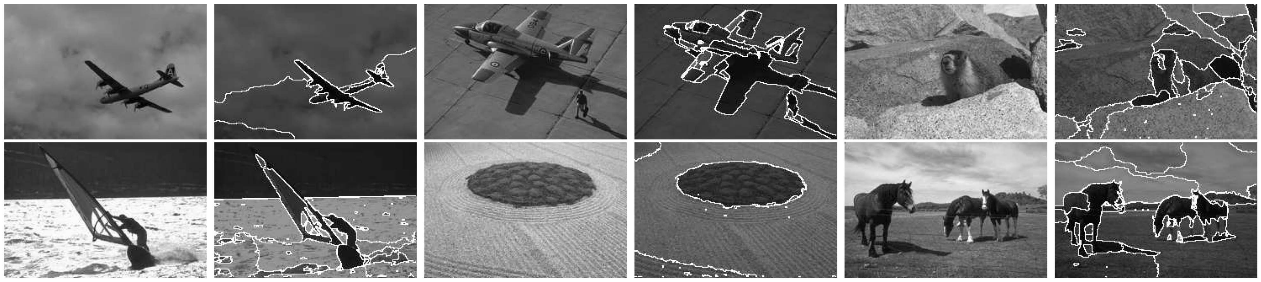


Fig. 14. Segmentation results obtained by our method. Image size 160×240.

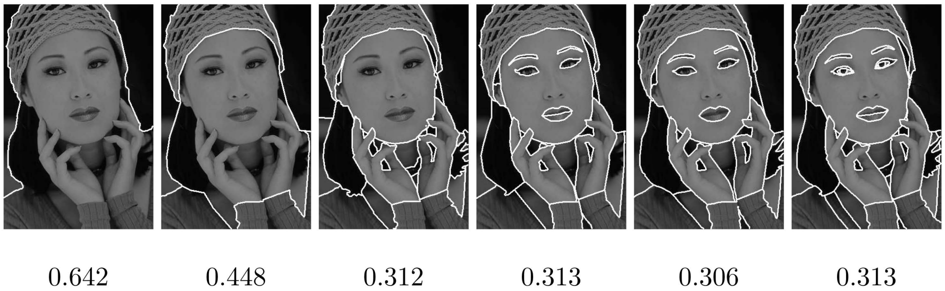


Fig. 15. Segmentation evaluation by the *average normalized* VoI distance between one segmentation and all of the manual segmentations together (the ground truth).

5 Experiments: Segmentation, Smoothing, Detection, and Tracking

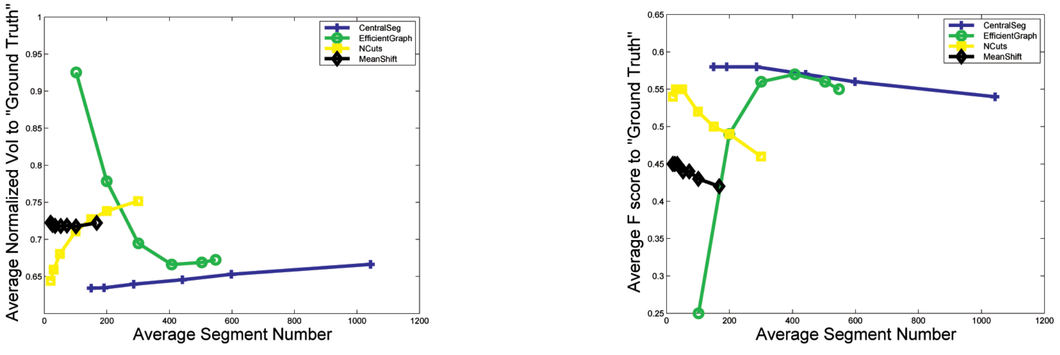
5.1 Image Segmentation

For a quantitative comparison, we use the Berkeley segmentation test set of one hundred 320×480 gray-scale images [29]. For faster computation, we resized to 160×240 (using Matlab bicubic smoothing). Since the standard boundary-consistency criterion in [30] isn't optimal for evaluating segmentations (see discussion in Section 1), especially for evaluating oversegmentations because many fake boundaries make finding reliable boundary correspondence more difficult, we use region consistency as our main criterion. We consider multiple human-labeled segmentations as giving a probability distribution for the "ground-truth," and evaluate segmentations by their averaged distance from this distribution using $\Delta(s_1, s_2) = V(s_1, s_2)/H(s_1, s_2)$. (Note this is *not* the $\Delta(s, F)$ minimized by our algorithm! A similar criterion has been applied for segmentation evaluation in [50].) The segmentation with the least averaged distance from the "ground-truth" distribution can be considered to be the closest to the central segmentation, therefore the best segmentation. Fig. 15 shows examples of applying this criterion to evaluate "ground-truth" segmentations.

We compared with the leading low level image segmentation algorithms, NC [40], the efficient graph (EG) approach [15] and mean shift segmentation [12]. The codes of the tested methods are from the corresponding authors. For objective comparison, we produce segmentations with different number of segments for different methods by changing the parameters. For our method, we used the Gaussian model (11) to compute pairwise affinities with $D = 2$. The only free parameter is the intensity standard deviation σ and we used σ between 35 and 300. Fig. 14 shows sample segmentations by our methods. For Ncuts, the varying parameter is the number of produced segments. EG has one free parameter K to control the region size in the output segmentation and we used K between 250 and 2,000. For MS, we use default parameters (spatial smoothing kernel 7 and range smoothing kernel 6.5) and intensity cues only. The free parameter is the minimum region size, which was varied between 15 and 200.

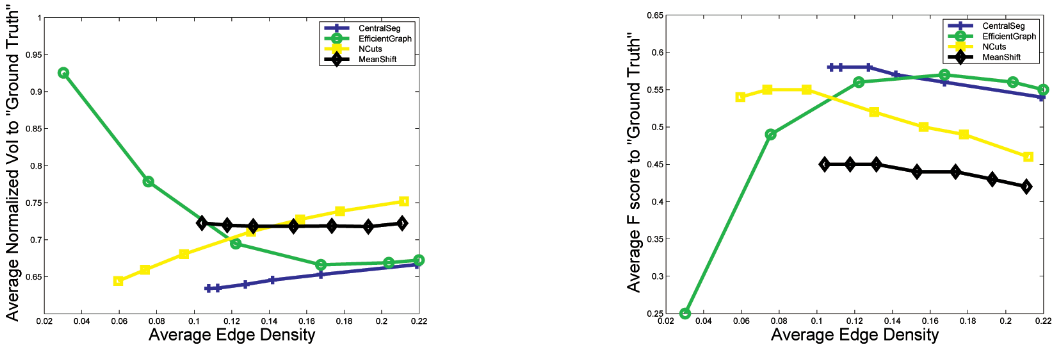
Fig. 16 compares the results using both our normalized VoI and Berkeley's F-measure [30]. To show more detail, we also present in Fig. 17 the same segmentation results in terms of edge density (percentage of edge pixels in the image) instead of number of segments. Overall, our method outperforms competing methods especially for our normalized VoI evaluation criterion. To directly compare the VoI and the F-measure, we plot the segmentation results using our algorithm with $\sigma = 300$, which produces about 150 regions on the test images on average, in Fig. 18. As we can see, VoI and F-measure have consistent performance. Since we compute affinities using a single value of σ across the

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.



(a) (b)

Fig. 16. Comparison in terms of number of regions. (a) Evaluated by normalized VoI; (b) evaluated by F-measure.



(a) (b)

Fig. 17. Comparison in terms of the edge density. (a) Evaluated by normalized VoI; (b) evaluated by F-measure.

whole image, we get some noisy small segments in regions with larger intensity variations. This can produce high edge densities and causes the slight dip in performance compared to EG [15] in Fig. 17. To correct this, one can adaptively estimate the optimal σ for different parts of the image, e.g., by the criterion in [15].

To investigate the stability and predictability of each segmentation algorithm, we use the boundary overlap ratio between segmentations containing fewer segments (edges) and segmentations containing more segments (edges). The stability and predictability of the segmentation algorithm can be evaluated for each image by:

$$c = 1 - \sum_{k \leq q} \frac{\sum_{e \in S_k \cap S_q} 1 / \sum_{e \in S_k \cup S_q} 1}{\sum_{k \leq q} 1}, \tag{29}$$

where S_k and S_q are the edgel sets for segmentations with k and q segments, respectively. This criterion has limitations because it always evaluates two segmentations as perfectly consistent if one is obtained by merging segments of the other. However, it is a reasonable estimate for segmentations that are obtained independently. By this criterion, the stability scores for our method, MS, NC, and EG are 0.3727, 0.4468, 0.6210, and 0.6266, respectively. Note that MS has better consistency than Ncuts and EG because all segmentations are obtained by merging from a single oversegmentation. The criterion is fair for the other three algorithms. Since we explicitly minimize the segmentation variance, our approach has the best stability and predictability.

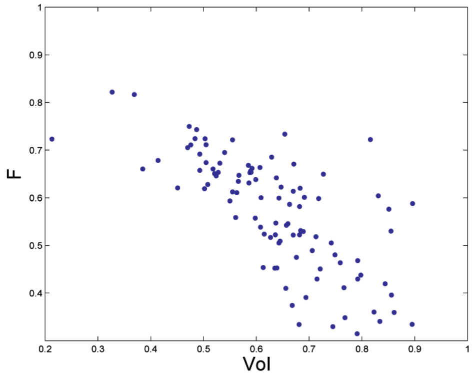


Fig. 18. The normalized VoI and F-measure show a good correlation.

5.2 Image Smoothing

We compare our smoothing results with Bilateral smoothing results. We ran both smoothing algorithms on the same test data used in our segmentation experiment. The smoothing radius is fixed to be $R = 2$ for both algorithms; 300 iterations are used. For the standard deviation of intensity in the Gaussian, we used 10, 20, 40, and 60 for each algorithm on each image separately. When the standard deviation used for the original image is also used for the candidate "most similar" images, our method generates smoothed images with enhanced boundary contrasts (see Fig. 13). To get boundary contrasts closer to the original

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

TABLE 2
Quantitative Results of Smoothing Results

intensity std	Ours	Bilateral	original
10	**0.8671**	0.8703	0.8839
20	**0.8318**	0.8341	0.8533
40	**0.8297**	0.8400	0.8476
60	**0.8582**	0.8785	0.8705

ones, for our smoothing algorithm, we used a standard deviation for the candidate images that is half of that used for the original image.

To examine how well the global structure is preserved/highlighted, we evaluate a smoothed image as in our segmentation evaluation by measuring the average distance between it and the "ground-truth" segmentations. The distance is measured using affinity matrices with radius of $D = 10$ and intensity standard deviation equal to that used for the original image. We also give the average distance between the original image and the "ground-truth" segmentations. See Table 2. In terms of highlighting the global structures, our smoothing consistently enhances the global structures and performs better than bilateral smoothing.

5.3 Detection

The "bag of features" (BoF) approach to recognition, e.g., [24], compares images according to the appearance (textures) of small patches. To cope with global variability, it takes a resolutely local approach, neglecting spatial layout almost completely. More recent work uses some layout information [25], or local-shape descriptors [32], [17], [41], or a combination of local appearance and shape [33], and gives improved performance especially for object detection. As discussed in Section 1, the local-shape approaches rely on supervised learning: To match an image, they must first build a descriptor of its contents from training images.

Our approach can approximately match rigid shapes for general images. It complements the local-appearance methods, and we could combine the approaches by treating ours as another semilocal descriptor for shapes that were approximately rigid over small regions. This would be the appropriate strategy for recognition of general articulated objects.

Here, we concentrate on testing our method, applying it for recognition/detection on its own. To demonstrate its robustness against global variability, we apply it as a *global* descriptor: We detect a class instance by measuring test images against a template containing the *whole* object plus context. For minimal learning, we detect objects simply by thresholding our matching score.

Again, we stress that our approach will work in this test only if the class instances have common underlying rigid structures, so that different instances are approximately self-similar. We test just on object categories with this property, namely, cars and faces. To match articulated objects such as animals and people, we would have to apply our approach more locally (at the limb level) and embed it in a more complete system.

TABLE 3
Detection Performances: Equal Error Rates (Percentage)

	Ours	[43]	[46]	[32]	[41]	[33]	[8]
Faces	98.1	93	92	96.4	97.2	99	99.1
Crear	100	98.2	86	97.7	98.2	100	99.1
Cside	90	–	–	85	–	93.8	88

We ran detection experiments on the UIUC car side-views [1] and the CalTech car-rear-view and face databases. The UIUC data contain 550 positive exemplars. We used their average affinity matrix as the car-side template, detecting cars simply by thresholding the normalized distance from the template. For the face and car-rear-view data with image size 200×132, we manually cut 10 positive examples from the first 10 images. Detection was done on the remaining images (106 car images and 424 face images) based on the average distance from these exemplars. The templates were scanned across the test images.

The final shape matching score is defined as the normalized distance to template subtracted by the normalized distance from the background pattern. The background pattern should be uniform everywhere and its affinity matrix is computed only considering the distance:

$$M_{bg}(m,n) = \begin{cases} exp\left(\frac{-d(m,n)^2}{2D^2}\right), & \text{if } d(m,n) \leq D; \\ 0, & \text{if } d(m,n) > D, \end{cases} \tag{30}$$

Instead of using (11), we estimated the affinities separately for each image based on its statistics. Given image F, let h_r^F be the empirical histogram of the absolute intensity difference between pixels at a relative distance r. To get the affinities, we normalize so that $h_r^F \in [0,1]$ and average over r. The gaussian in (11) is replaced by:

$$h^F(i) = D^{-1} \sum_{r=1:D} h_r^F(i), \tag{31}$$

where i is intensity. Normalizing $h_r^F \in [0,1]$ gives higher weights to closer pixels. The choice for the cutoff D should reflect the scale of the object. For the car-side, car-rear, and face data, we used $D = 2$, 7, and 5, respectively.

For comparison, we implemented two other approaches [43], [46] that match images by their affinity matrices. As expected, our method gives better results than [43], [46] (see Table 3). Our results are also better than the intermediate-level grouping approaches [32], [41], which learn local-shape descriptors from training images, assuming, as we do, that the object is delineated by a bounding box. We also compared with the pyramid of histogram of gradients (PHoG) approach [8]. The PHoG code is from the corresponding author. We use four scales with 40 bins on a 360-degree angle histogram, which has been shown to give the optimal performance. For the face and car-rear-view data, the same detection routine as ours is used for PHoG. For the car-side-view data, an SVM classifier is trained with the provided training images for detection. As we can see, as a global shape descriptor, PHoG has comparable results as ours and is also better than local-shape methods. This shows the importance of using global shapes in shape matching. Another approach, [33], does better on car-side and slightly better on faces, but unlike us uses both shape and appearance. Considering all reported results on these data, our performance on faces and car rear

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

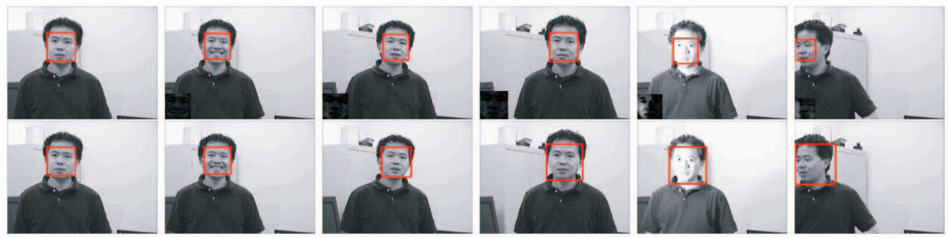


Fig. 19. Tracking on the Ming sequence. Top: ours; bottom: PHoG. From left to right are frames 1, 300, 600, 900, 1200, and 1500.

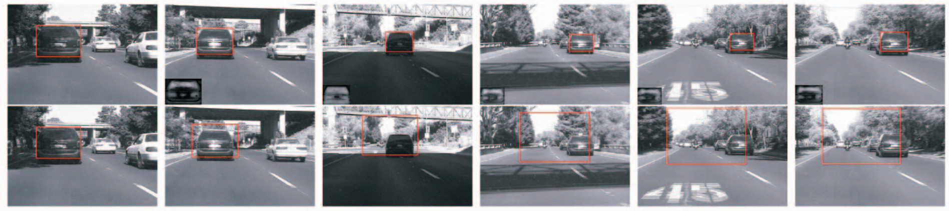


Fig. 20. Tracking on the Car4 sequence. Top: ours; bottom: PHoG. From left to right are frames 1, 110, 220, 330, 440, and 550.

views is close to the state of the art with many fewer exemplars, with slightly worse results for the car-side data. The latter contains images with significant global occlusion, so our global matching strategy does not perform as well.

5.4 Tracking

We next present a simple application of our matching measure to tracking. Our motivation for this tracking experiment is not to develop a complete, state-of-the-art tracking algorithm (for instance, we have no mechanism for recovering from a lost target), but as another way to show the power of our matching method. Unlike many shape-based trackers that use global active contours, we don't require a good initialization and, unlike approaches that weaken the shape representation to make it robust to shape distortion, for instance, representing the object in terms of histograms over gradients, e.g., [13], we track the detailed shape.

Since our matching measure has a built-in robustness to shape distortions, we can implement tracking essentially as simple template matching. The algorithm tracks an object by moving the tracker to the best shape match in each frame. For demonstration, we use brute force search to find the best location and the optimal scale. For robustness against occlusion and appearance changes, we include history into the template, updating its affinity matrix description by $M_F^{(n)} = ((w-1)M_F^{(n-1)} + M_{F(n-1)})/w$, where $M_{F(n-1)}$ is the affinity matrix for image $n-1$ alone, w is a weight parameter, and we use $M_F^{(n)}$ for matching. Figs. 19 and 20 use weight $w = 20$.

Our results on outdoor and indoor sequences from [37], show our method's robustness to deformation, camera motion, and changes in illumination, scale and pose.[5] For faster speed, we resize the tracking region into smaller size (no larger than 50×50) via the bicubic function. We compute the dynamic affinity function based upon the neighbor image area that is centered at the current location and has an area three times larger than the tracked region. Our Matlab brute force search implementation takes ~ 15 seconds to process one frame with a 3,200 HZ AMD.

For comparison, we implement PHoG (four scales, $\mathrm{angle} = 360$, $\mathrm{bin} = 40$) based shape tracking using the same routine but without resizing the tracked regions (resizing gave worse results). Even though PHoG uses multiscale shapes and our method only uses single coarse scale shapes, we track objects more accurately in both location and scale than PHoG. PHoG has less robustness to illumination variation since it relies on locally determined edge orientations, whereas our method uses globally computed segmentations. In summary, the test shows that our segmentation-based shape matching is more accurate than PHoG in comparing similar shapes with small deformations in different lighting conditions. Our tracking results also compare well to the results in [37].

6 Conclusions

We propose and give solutions to use image segmentations for approximately rigid shape matching. Our method can match curves without correspondence over large-scale image regions, with good robustness to local-shape variations and occlusion. It can exploit global shape structures which are more distinctive than local edge features. To address the unreliability of image segmentations, we describe a closed form approximation to an average over all segmentations. This technique makes segmentation-based shape matching more practical and more efficient.

As a closed loop, our approximation technique gives new algorithms for image segmentation. In addition, our work offers a new perspective to image smoothing. A new edge-preserving smoothing algorithm is presented to smooth images according to their globally optimal structures. For segmentation and smoothing evaluation, we use an objective criterion that compares computed segmentations/smoothed images to "ground-truth" segmentations produced by humans.

Finally, since our approach compares signals based on their internal structure, it can easily adapt to matching signals from different modalities, e.g., images taken in

[5] We also tested on PETS 2007. See [48] and our Web page for further tracking results.

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

different frequency bands, or visual images matched to sonar/radar data.

Appendix A

We prove that $\Delta(x,y) = \frac{V(x,y)}{H(x,y)}$ is a metric for segmentations by showing:

- **Positivity**: $\Delta(x,y) \geq 0$ and $\Delta(x,y) = 0$ *iff* $x = y$.
- **Symmetry**: $\Delta(x,y) = \Delta(y,x)$.
- **Triangle inequality**: $\Delta(x,z) + \Delta(y,z) \geq \Delta(x,y)$.

It is easy to see that the first two properties hold. We show that the third property also holds. Since $H(x,z), H(z,y), H(x,y) \leq H(x,y,z)$, we only need to show that $\frac{V(x,z)+V(z,y)}{H(x,y,z)} \geq \frac{V(x,y)}{H(x,y)}$ or:

$$\begin{aligned}
&[V(x,z) + V(z,y)]H(x,y)\\
&\quad = [2H(x,z) + 2H(y \mid z) - H(x) - H(y)]H(x,y)\\
&\quad \geq [2H(x,z) + 2H(y \mid x,z) - H(x) - H(y)]H(x,y)\\
&\quad \geq [2H(x,y) - H(x) - H(y)]H(x,y,z)\\
&\quad = V(x,y)H(x,y,z).
\end{aligned}$$

Appendix B

We prove that V_ω obeys the triangle inequality. For any images F_1, F_2, and F_3:

$$\begin{aligned}
V_\omega(F_1,F_2) &= \sum_{s_3 \in S} p(s_3 \mid F_3) \sum_{s_1,s_2 \in S} p(s_1 \mid F_1)p(s_2 \mid F_2)V(s_1,s_2)\\
&\leq \sum_{s_1,s_2,s_3 \in S} p(s_1 \mid F_1)p(s_2 \mid F_2)p(s_3 \mid F_3)[V(s_1,s_3)\\
&\quad + V(s_2,s_3)] = V_\omega(F_1,F_3) + V_\omega(F_2,F_3).
\end{aligned}$$

Appendix C

We prove that Δ_ω obeys the triangle inequality. Since $V_\omega(F_1,F_2) \leq V_\omega(F_1,F_3) + V_\omega(F_2,F_3)$, it is easy to see that the triangle inequality holds when $H_\omega(F_1,F_2) \geq H_\omega(F_1,F_3)$, $H_\omega(F_2,F_3)$. We prove that the triangle inequality also holds for 1) $H_\omega(F_1,F_3) < H_\omega(F_1,F_2) \leq H_\omega(F_2,F_3)$ and 2) $H_\omega(F_1,F_2) < H_\omega(F_1,F_3), H_\omega(F_2,F_3)$. We give proof for case 1 and it is easy to extend the proof for case 2.

$$\begin{aligned}
\Delta_\omega(F_1,F_3) + \Delta_\omega(F_2,F_3) &\geq \frac{V_\omega(F_1,F_3)}{H_\omega(F_1,F_2)} + \frac{V_\omega(F_2,F_3)}{H_\omega(F_2,F_3)}\\
&= \frac{2H_\omega(F_1,F_3)}{H_\omega(F_1,F_2)} - \frac{H_\omega(F_1) + H_\omega(F_3)}{H_\omega(F_1,F_2)}\\
&\quad + 2 - \frac{V_\omega(F_2) + V_\omega(F_3)}{H_\omega(F_2,F_3)}\\
&\geq 2 - \frac{H_\omega(F_1)}{H_\omega(F_1,F_2)} - \frac{H_\omega(F_2)}{H_\omega(F_2,F_3)}\\
&\geq 2 - \frac{V_\omega(F_1) + V_\omega(F_2)}{H_\omega(F_1,F_2)}\\
&= \Delta_\omega(F_1,F_2).
\end{aligned}$$

Acknowledgments

The authors would like to thank the anonymous reviewers for their critical insights.

References

[1] S. Agarwal and D. Roth, "Learning a Sparse Representation for Object Detection," *Proc. European Conf. Computer Vision,* 2002.

[2] N. Ahuja and S. Todorovic, "Learning the Taxonomy and Models of Categories Present in Arbitrary Images," *Proc. Int'l Conf. Computer Vision,* 2007.

[3] H. Barrow, J. Tenenbaum, R. Bolles, and H. Wolf, "Parametric Correspondence and Chamfer Matching: Two New Techniques for Image Matching," *Proc. Fifth Int'l Joint Conf. Artificial Intelligence,* 1977.

[4] R. Basri and D. Jacobs, "Recognition Using Region Correspondences," *Int'l J. Computer Vision,* vol. 25, pp. 145-166, 1997.

[5] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 4, pp. 509-522, Apr. 2002.

[6] A. Berg and J. Malik, "Geometric Blur for Template Matching," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2001.

[7] A. Berg, T. Berg, and J. Malik, "Shape Matching and Object Recognition Using Low Distortion Correspondences," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2005.

[8] A. Bosch, A. Zisserman, and X. Munoz, "Representing Shape with a Spatial Pyramid Kernel," *Proc. ACM Int'l Conf. Image and Video Retrieval,* 2007.

[9] E. Borenstein and S. Ullman, "Combined Top-Down/Bottom-Up Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 12, pp. 2109-2125, Dec. 2008.

[10] Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 11, pp. 1222-1239, Nov. 2001.

[11] G. Charpiat, O. Faugeras, and R. Keriven, "Shape Statistics for Image Segmentation with Prior," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2007.

[12] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach towards Feature Space Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 5, pp. 603-619, May 2002.

[13] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2005.

[14] H.Q. Dinh and S. Kropac, "Multi-Resolution Spin Images," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2006.

[15] P. Felzenszwalb and D. Huttenlocher, "Efficient Graph-Based Image Segmentation," *Int'l J. Computer Vision,* vol. 59, no. 2, pp. 109-131, 2004.

[16] V. Ferrari, T. Tuytelaars, and L. Van Cool, "Simultaneous Object Recognition and Segmentation from Single or Multiple Model Views," *Int'l J. Computer Vision,* vol. 67, no. 2, pp. 159-188, 2006.

[17] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid, "Groups of Adjacent Contour Segments for Object Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 1, pp. 36-51, Jan. 2008.

[18] Y. Gdalyahu, D. Weinshall, and M. Werman, "Self Organization in Vision: Stochastic Clustering for Image Segmentation, Perceptual Grouping, and Image Database," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 10, pp. 1053-1074, Oct. 2001.

[19] C.-Y. Huang, O.I. Camps, and T. Kanungo, "Object Recognition Using Appearance-Based Parts and Relations," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 1997.

[20] D. Huttenlocher, G. Klanderman, and W. Rucklidge, "Comparing Images Using the Hausdorff Distance," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 15, no. 9, pp. 850-863, Sept. 1993.

[21] A.E. Johnson and M. Hebert, "Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, no. 5, pp. 433-449, May 1999.

[22] N. Jojic and Y. Caspi, "Capturing Image Structure with Probabilistic Index Maps," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2004.

[23] S. Joshi, B. Davis, M. Jomier, and G. Gerig, "Unbiased Diffeomorphic Atlas Construction for Computational Anatomy," *NeuroImage,* vol. 23, pp. 15-160, 2004.

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.

[24] F. Jurie and B. Triggs, "Creating Efficient Codebooks for Visual Recognition," *Proc. Int'l Conf. Computer Vision,* 2005.

[25] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2006.

[26] B. Leibe, A. Leonardis, and B. Schiele, "Combined Object Categorization and Segmentation with an Implicit Shape Model," *Proc. European Conf. Computer Vision Workshop Statistical Learning in Computer Vision,* 2004.

[27] A. Levin and Y. Weiss, "Learning to Combine Bottom-Up and Top-Down Segmentation," *Int'l J. Computer Vision,* vol. 81, no. 1, pp. 105-118, 2009.

[28] D. Lowe, "Distinctive Image Features from Scale Invariant Keypoints," *Int'l J. Computer Vision,* vol. 60, pp. 91-110, 2004.

[29] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A Database of Human Segmented Natural Images and Its Applications to Evaluating Segmentation Algorithms and Measuring Ecological Statistics," *Proc. Int'l Conf. Computer Vision,* 2001.

[30] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "Learning to Detect Natural Image Boundaries Using Local Brightness, Color, and Texture Cues," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 5, pp. 530-549, May 2004.

[31] M. Meila, "Comparing Clusterings by the Variation of Information," *Proc. Conf. Learning Theory,* 2003.

[32] A. Opelt, A. Pinz, and A. Zisserman, "A Boundary-Fragment-Model for Object Detection," *Proc. European Conf. Computer Vision,* 2006.

[33] A. Opelt, A. Pinz, and A. Zisserman, "Fusing Shape and Appearance Information for Object Category Detection," *Proc. British Machine Vision Conf.,* 2006.

[34] P. Perona and J. Malik, "Scale-Space and Edge Detection Using Anisotropic Diffusion," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 12, no. 7, pp. 629-639, July 1990.

[35] X. Ren and J. Malik, "Learning a Classification Model for Segmentation," *Proc. Int'l Conf. Computer Vision,* 2003.

[36] D. Russakoff, C. Tomasi, T. Rohlfing, and C. Maurer, "Image Similarity Using Mutual Information of Regions," *Proc. European Conf. Computer Vision,* 2004.

[37] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental Learning for Robust Visual Tracking," *Int'l J. Computer Vision,* vol. 77, pp. 125-141, 2007.

[38] B. Russell, A. Efros, J. Sivic, W. Freeman, and A. Zisserman, "Using Multiple Segmentations to Discover Objects and Their Extent in Image Collections," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2006.

[39] E. Sharon, M. Galun, D. Sharon, R. Basri, and A. Brandt, "Hierarchy and Adaptivity in Segmenting Visual Scenes," *Nature,* vol. 442, pp. 810-813, 2006.

[40] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 8, pp. 888-905, Aug. 2000.

[41] J. Shotton, A. Blake, and R. Cipolla, "Multi-Scale Categorical Object Recognition Using Contour Fragments," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 7, pp. 1270-1281, July 2008.

[42] K. Siddiqi, A. Shokoufandeh, S. Dickinson, and S. Zucker, "Shock Graphs and Shape Matching," *Int'l J. Computer Vision,* vol. 35, pp. 13-32, 1999.

[43] C. Stauffer and E. Grimson, "Similarity Templates for Detection and Recognition," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2001.

[44] H. Sundar, D. Silver, N. Gagvani, and S. Dickinson, "Skeleton Based Shape Matching and Retrieval," *Proc. Shape Modeling Int'l,* 2003.

[45] C. Tomasi and R. Manduchi, "Bilateral Filtering for Gray and Color Images," *Proc. Int'l Conf. Computer Vision,* 1998.

[46] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward Objective Evaluation of Image Segmentation Algorithms," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 6, pp. 929-944, June 2007.

[47] P. Viola and W.M. Wells III, "Alignment by Maximization of Mutual Information," *Int'l J. Computer Vision,* vol. 24, no. 2, pp. 137-154, 1997.

[48] H. Wang and J. Oliensis, "Shape Matching by Segmentation Averaging," *Proc. European Conf. Computer Vision,* 2008.

[49] J. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 9, pp. 947-963, Sept. 2001.

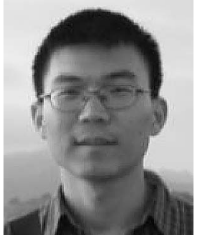
[50] A. Yang, J. Wright, Y. Ma, and S. Sastry, "Unsupervised Segmentation of Natural Images via Lossy Data Compression," *Computer Vision and Image Understanding,* vol. 110, no. 2, pp. 212-225, 2008.

[51] W.E. Young and R.H. Trent, "Geometric Mean Approximation of Individual Security and Portfolio Performance," *J. Financial and Quantitative Analysis,* vol. 4, pp. 179-199, 1969.

[52] S.X. Yu, R. Gross, and J. Shi, "Concurrent Object Recognition and Segmentation by Graph Partitioning," *Proc. Neural Information Processing Systems,* 2002.

[53] L. Zhu, C. Lin, H. Huang, Y. Chen, and A. Yuille, "Unsupervised Structure Learning: Hierarchical Recursive Composition, Suspicious Coincidence and Competitive Exclusion," *Proc. European Conf. Computer Vision,* 2008.

[54] S.C. Zhu and A. Yuille, "FORMS: A Flexible Object Recognition and Modelling System," *Int'l J. Computer Vision,* vol. 20, pp. 187-212, 1996.



Hongzhi Wang received the bachelor's and master's degrees in computer science from the University of Science and Technology, Beijing, in 2000 and 2003, respectively. In 2008, he received the PhD degree from the Stevens Institute of Technology. His research interests are in perceptual organization, object recognition, and medical image analysis. Currently, he is a post doctoral researcher in the Penn Image Computing and Science Lab at the University of Pennsylvania. He is a member of the IEEE.

John Oliensis received the PhD degree in theoretical particles physics from the University of Chicago and carried out research in physics at Princeton University, the Fermi National Accelerator Laboratory, and the Argonne National Laboratory. He began research in computer vision in 1988, joining the University of Massachusetts at Amherst as a member of the research faculty. From 1994 to 2003, he was a research scientist at the NEC Research Institute, where he organized three workshops bringing together researchers in computer vision, human vision, neuroscience, and learning. Since 2003, he has been an associate professor in the Computer Science Department at the Stevens Institute of Technology. His interests include the estimation of object shape from images, perceptual organization, the recognition of objects, and human vision. He is a senior member of the IEEE and an associate editor of the *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.

Authorized licensed use limited to: The University of Arizona. Downloaded on February 01,2023 at 20:10:18 UTC from IEEE Xplore. Restrictions apply.