Principal Curvature-Based Region Detector for Object Recognition

Hongli Deng, Wei Zhang, Eric Mortensen, Thomas Dietterich
Oregon State University
Corvallis, OR, 97330
{deng,zhangwe,enm,tgd}@eecs.oregonstate.edu

Linda Shapiro
University of Washington
Seattle, WA 98195
shapiro@cs.washington.edu

Abstract

This paper presents a new structure-based interest region detector called Principal Curvature-Based Regions (PCBR) which we use for object class recognition. The PCBR interest operator detects stable watershed regions within the multi-scale principal curvature image. To detect robust watershed regions, we "clean" a principal curvature image by combining a grayscale morphological close with our new "eigenvector flow" hysteresis threshold. Robustness across scales is achieved by selecting the maximally stable regions across consecutive scales. PCBR typically detects distinctive patterns distributed evenly on the objects and it shows significant robustness to local intensity perturbations and intra-class variations. We evaluate PCBR both qualitatively (through visual inspection) and quantitatively (by measuring repeatability and classification accuracy in real-world object-class recognition problems). Experiments on different benchmark datasets show that PCBR is comparable or superior to state-of-art detectors for both feature matching and object recognition. Moreover, we demonstrate the application of PCBR to symmetry detection.

1. Introduction

In many object recognition tasks, within-class changes in pose, lighting, color, and texture can cause considerable variation in local intensities. Consequently, local intensity no longer provides a stable detection cue. As such, intensity-based interest operators (e.g., Harris, Kadir)–and the object recognition systems based on them–often fail to identify discriminative features. An alternative to local intensity cues is to capture semi-local structural cues such as edges and curvilinear shapes [25]. These structural cues tend to be more robust to intensity, color, and pose variations. As such, they provide the basis for a more stable interest operator, which in turn improves object recognition accuracy. This paper introduces a new detector that exploits curvilinear structures to reliably detect interesting regions. The detector, called the Principal Curvature-Based Region (PCBR) detector, identifies stable watershed regions within the multi-scale principal curvature image.

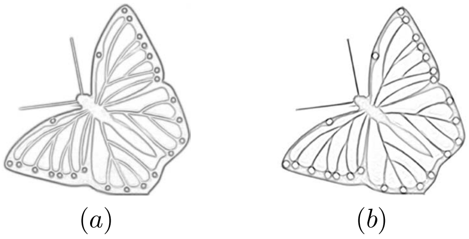


Figure 1. Comparison of the gradient magnitude and principal curvature responses of the image in Figure 2(a). (a) Gradient magnitude response. (b) Principal curvature response.

Curvilinear structures are lines (either curved or straight) such as roads in aerial or satellite images or blood vessels in medical scans. These curvilinear structures can be detected over a range of viewpoints, scales, and illumination changes. The PCBR detector employs the first steps of Steger's curvilinear detector algorithm [25]. It forms an image of the maximum or minimum eigenvalue of the Hessian matrix at each pixel. We call this the principal curvature image, as it measures the principal curvature of the image intensity surface. This process generates a single response for both lines and edges, producing a clearer structural sketch of an image than is usually provided by the gradient magnitude image (see Fig. 1).

We develop a process that detects structural regions efficiently and robustly using the watershed transform of the principal curvature image across scale space. The watershed algorithm provides a more efficient mechanism for defining structural regions than previous methods that fit circles, ellipses, and parallelograms [8, 27]. To improve the watershed's robustness to noise and other small image perturbations, we first "clean" the principal curvature image with a grayscale morphological close operation followed by a new hysteresis thresholding method based on local eigenvector flow. The watershed transform is then applied to the cleaned principal curvature image and the resulting watershed regions (i.e., the catchment basins) define the PCBR regions. To achieve robust detections across multiple scales, the watershed is applied to the maxima of three consecutive images in the principal curvature scale space–similar to local scale-space extrema used by Lowe

[13], Mikolajczyk and Schmidt [17], and others–and we further search for stable PCBR regions across consecutive scales–an idea adapted from the stable regions detected across multiple threshold levels used by the MSER detector [15]. While PCBR shares similar ideas with previous detectors, it represents a very different approach to detecting interest regions. Many prior intensity-based detectors search for points with distinctive local differential geometry, such as corners, while ignoring image features such as lines and edges. Conversely, PCBR utilizes line and edge features to construct structural interest regions. Compared to MSER, PCBR differs two important aspects. First, MSER does not analyze regions in scale space, so it does not provide different levels of region abstraction. Second, MSER's intensity-based threshold process cannot overcome local intensity variations within regions. PCBR, however, overcomes this difficulty by focusing on region boundaries rather than the appearance of region interiors.

This work makes two contributions. First, we develop a new interest operator that utilizes principal curvature to extract robust and invariant region structures based on both edge and curvilinear features. Second, we introduce an enhanced principle-curvature-based watershed segmentation and robust region selection process that is robust to intra-class variations and is more efficient than previous structure-based detectors. We demonstrate the value of our PCBR detector by applying it to object-class recognition problems and symmetry detection.

2. Related Work

Interest operators can typically be classified into two categories: intensity-based detectors and structure-based detectors [19]. Intensity-based detectors depend on analyzing local differential geometry or intensity patterns to find points or regions that satisfy some uniqueness and stability criteria. The Harris corner detector [7] finds points or pixels where both eigenvalues of the second moment matrix are large by evaluating the "Harris measure". The Harris-affine and Hessian-affine detectors [17, 18] compute maximum determinants of the second moment matrix and the Hessian matrix respectively across scale space and then apply Laplacian-based characteristic scale selection [11] and second-moment-matrix-based shape adaptation [12, 2]. MSER [15] uses a threshold selection process to detect stable regions that are either brighter or darker than the surrounding region. SIFT (i.e., the DoG extrema detector used by Lowe in [13]) finds local extrema across three consecutive difference-of-Gaussian scales and then removes spurious detections via a DoG-response threshold followed by a Harris-like metric to eliminate edge detections. Kadir's salient region detector [9] calculates the entropy of the probability density function (PDF) of intensity values over various scales to find regions with entropy extrema. Other intensity-based detectors include SUSAN [24], intensity extrema-based regions (IBR) [26], and the work of Moravec [21] and Beaudet [3].

Structure-based detectors depend on structural image features such as lines, edges, curves, etc. to define interest points or regions. These detectors tend to be very computationally expensive and typically depend on reliable prior detection of structural features. Early structure-based detectors analyze various 2D curves such as the curvature primal sketch or B-splines extracted from edges, ridges, troughs, etc. and then selected high curvature points, line or curve intersections, corners, ends, bumps, and dents as interest points [1, 16, 5, 23, 20]. Tuytelaar's edge-based region (EBR) detector [27] fits a parallelogram defined by Harris corner point and points on two adjacent edge contours (extracted by the Canny detector [4]). Scale-invariant shape features (SISF) [8] detects circles at different locations and scales by evaluating salient convex arrangements of Canny edges based on a measure that maximizes how well a circle is supported by surrounding edges.

3. Principal curvature-based Region Detector

3.1. Principal Curvature Image

Two types of structures have high curvature in one direction and low curvature in the orthogonal direction: lines (i.e., straight or nearly straight curvilinear features) and edges. Viewing an image as an intensity surface, the curvilinear structures correspond to ridges and valleys of this surface. The local shape characteristics of the surface at a particular point can be described by the Hessian matrix,

$$\mathbf{H}(\mathbf{x}, \sigma_D) = \begin{bmatrix} I_{xx}(\mathbf{x}, \sigma_D) & I_{xy}(\mathbf{x}, \sigma_D) \\ I_{xy}(\mathbf{x}, \sigma_D) & I_{yy}(\mathbf{x}, \sigma_D) \end{bmatrix}, \quad (1)$$

where I_{xx}, I_{xy} and I_{yy} are the second-order partial derivatives of the image evaluated at the point $\mathbf{x}$ and σ_D is the Gaussian scale of the partial derivatives.

We note that both the Hessian matrix and the related second moment matrix have been applied in several other interest operators (e.g., the Harris [7], Harris-affine [19], and Hessian-affine [18] detectors) to find image positions where the local image geometry is changing in more than one direction. Likewise, Lowe's maximal difference-of-Gaussian (DoG) detector [13] also uses components of the Hessian matrix (or at least approximates the sum of the diagonal elements) to find points of interest. However, our PCBR detector is quite different from these other methods and is complementary to them. Rather than finding extremal "points", our detector applies the watershed algorithm to ridges, valleys, and cliffs of the image principal-curvature surface to find "regions". As with extremal points, the ridges, valleys, and cliffs can be detected over a range of viewpoints, scales, and appearance changes.

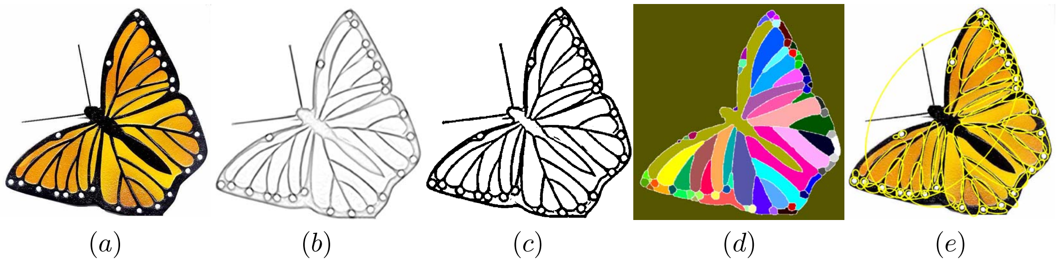


Figure 2. Interest Regions detected by the PCBR detector. (a) Original butterfly image. (b) Principal curvature and (c) cleaned binary images. (d) Watershed regions. (e) Detected regions represented by ellipses.

Many previous interest point detectors [7, 19, 18] apply the Harris measure (or a similar metric [13]) to determine a point's saliency. The Harris measure is given by $det(\mathbf{A}) - k \cdot tr^2(\mathbf{A}) > threshold$ where det is the determinant, tr is the trace, and the matrix $\mathbf{A}$ is either the Hessian matrix or the second moment matrix. One advantage of the Harris metric is that it does not require explicit computation of the eigenvalues. However, computing the eigenvalues for a 2×2 matrix requires only a single Jacobi rotation to eliminate the off-diagonal term, I_{xy}, as noted by Steger [25].

The Harris measure produces low values for "long" structures that have a small first or second derivative in one particular direction. Our PCBR detector compliments previous interest point detectors in that we abandon the Harris measure and exploit those very long structures as detection cues. The principal curvature image is given by either

$$P(\mathbf{x}) = \max(\lambda_1(\mathbf{x}), 0) \tag{2}$$

or

$$P(\mathbf{x}) = \min(\lambda_2(\mathbf{x}), 0) \tag{3}$$

where $\lambda_1(\mathbf{x})$ and $\lambda_2(\mathbf{x})$ are the maximum and minimum eigenvalues, respectively, of H at $\mathbf{x}$. Eq. 2 provides a high response only for dark lines on a light background (or on the dark side of edges) while Eq. 3 is used to detect light lines against a darker background.

Like SIFT [13] and other detectors, principal curvature images are calculated in scale space. We first double the size of the original image to produce our initial image, I_{11}, and then produce increasingly Gaussian smoothed images, I_{1j}, with scales of $\sigma = k^{j-1}$ where $k = 2^{1/3}$ and $j = 2..6$. This set of images spans the first octave consisting of six images, I_{11} to I_{16}. Image I_{14} is down sampled to half its size to produce image I_{21}, which becomes the first image in the second octave. We apply the same smoothing process to build the second octave, and continue to create a total of $n = log2(min(w,h)) - 3$ octaves, where w and h are the width and height of the doubled image, respectively. Finally, we calculate a principal curvature image, P_{ij}, for each smoothed image by computing the maximum eigenvalue (Eq. 2) of the Hessian matrix at each pixel. For computational efficiency, each smoothed image and its corresponding Hessian image is computed from the previous smoothed image using an incremental Gaussian scale.

Given the principal curvature scale space images, we calculate the maximum curvature over each set of three consecutive principal curvature images to form the following set of four images in each of the n octaves:

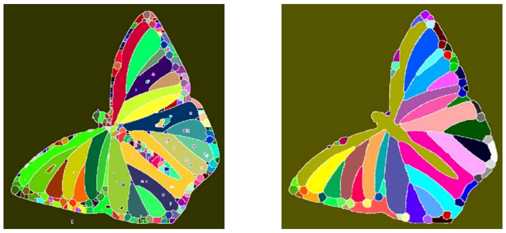
$$\begin{array}{cccc} MP_{12} & MP_{13} & MP_{14} & MP_{15} \\ MP_{22} & MP_{23} & MP_{24} & MP_{25} \\ \ldots & & & \\ MP_{n2} & MP_{n3} & MP_{n4} & MP_{n5} \end{array} \tag{4}$$

where $MP_{ij} = \max(P_{ij-1}, P_{ij}, P_{ij+1})$.

Figure 2(b) shows one of the maximum curvature images, MP, created by maximizing the principal curvature at each pixel over three consecutive principal curvature images. From these maximum principal curvature images we find the stable regions via our watershed algorithm.

3.2. Enhanced Watershed Regions Detections

The watershed transform is an efficient technique that is widely employed for image segmentation. It is normally applied either to an intensity image directly or to the gradient magnitude of an image. We instead apply the watershed transform to the principal curvature image. However, the watershed transform is sensitive to noise (and other small perturbations) in the intensity image. A consequence of this is that the small image variations form local minima that result in many, small watershed regions. Figure 3(a) shows the over-segmentation results when the watershed algorithm is applied directly to the principal curvature image in Figure 2(b)). To achieve a more stable watershed segmentation, we first apply a grayscale morphological closing followed by hysteresis thresholding. The grayscale morphological closing operation is defined as $f \bullet b = (f \oplus b) \ominus b$ where f is the image MP from Eq. 4, b is a 5×5 disk-shaped structuring element, and $\oplus$ and $\ominus$ are the grayscale dilation and erosion, respectively. The closing operation removes small "potholes" in the principal curvature terrain, thus eliminating many local minima that result from noise and that would otherwise produce watershed catchment basins.



(a) (b)

Figure 3. (a) Watershed segmentation of original principal curvature image (Fig. 2b). (b) Watershed segmentation of the "clean" principal curvature image (Fig. 2c).

Beyond the small (in terms of area of influence) local minima, there are other variations that have larger zones of influence and that are not reclaimed by the morphological closing. To further eliminate spurious or unstable watershed regions, we threshold the principal curvature image to create a clean, binarized principal curvature image. However, rather than apply a straight threshold or even hysteresis thresholding–both of which can still miss weak image structures–we apply a more robust eigenvector-guided hysteresis thresholding to help link structural cues and remove perturbations. Since the eigenvalues of the Hessian matrix are directly related to the signal strength (i.e., the line or edge contrast), the principal curvature image may, at times, become weak due to low contrast portions of an edge or curvilinear structure. These low contrast segments may potentially cause gaps in the thresholded principal curvature image, which in turn cause watershed regions to merge that should otherwise be separate. However, the directions of the eigenvectors provide a strong indication of where curvilinear structures appear and they are more robust to these intensity perturbations than is the eigenvalue magnitude.

In eigenvector-flow hysteresis thresholding, there are two thresholds (high and low) just as in traditional hysteresis thresholding. The high threshold (set at 0.04) indicates a strong principal curvature response. Pixels with a strong response act as seeds that expand to include connected pixels that are above the low threshold. Unlike traditional hysteresis thresholding, our low threshold is a function of the support that each pixel's major eigenvector receives from neighboring pixels. Each pixel's low threshold is set by comparing the direction of the major (or minor) eigenvector to the direction of the 8 adjacent pixels' major (or minor) eigenvectors. This can be done by taking the absolute value of the inner product of a pixel's normalized eigenvector with that of each neighbor. If the average dot product over all neighbors is high enough, we set the low-to-high threshold ratio to 0.2 (for a low threshold of $0.04 \cdot 0.2 = 0.008$); otherwise the low-to-high ratio is set to 0.7 (giving a low threshold of 0.028). The threshold values are based on visual inspection of detection results on many images.

Figure 4 illustrates how the eigenvector flow supports an otherwise weak region. The red arrows are the major eigenvectors, and the yellow arrows are the minor eigenvectors. To improve visibility, we draw them at every fourth pixel. At the point indicated by the large white arrow, we see that the eigenvalue magnitudes are small and the ridge there is almost invisible. Nonetheless, the directions of the eigenvectors are quite uniform. This eigenvector-based active thresholding process yields better performance in building continuous ridges and in handling perturbations, which results in more stable regions (Fig. 3(b)).

The final step is to perform the watershed transform on the clean binary image (Fig. 2(c)). Since the image is binary, all black (or 0-valued) pixels become catchment basins and the midlines of the thresholded white ridge pixels become watershed lines if they separate two distinct catchment basins. To define the interest regions of the PCBR detector in one scale, the resulting segmented regions are fit with ellipses, via PCA, that have the same second-moment as the watershed regions (Fig. 2(e)).

3.3. Stable Regions Across Scale

Computing the maximum principal curvature image (as in Eq. 4) is only one way to achieve stable region detections. To further improve robustness, we adopt a key idea from MSER and keep only those regions that can be detected in at least three consecutive scales. Similar to the process of selecting stable regions via thresholding in MSER, we select regions that are stable across local scale changes. To achieve this, we compute the overlap error of the detected regions across each triplet of consecutive scales in every octave. The overlap error is calculated the same as in [19].

Overlapping regions that are detected at different scales normally exhibit some variation. This variation is valuable for object recognition because it provides multiple descriptions of the same pattern. An object category normally exhibits large within-class variation in the same area. Since detectors have difficulty locating the interest area accurately, rather than attempt to detect the "correct" region and extract a single descriptor vector, it is better to extract multiple descriptors for several overlapping regions, provided that these descriptors are handled properly by the classifier.

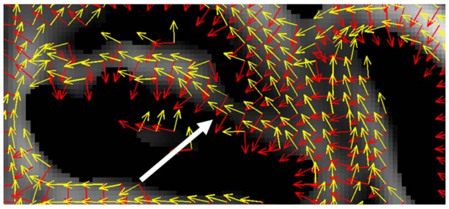


Figure 4. Illustration of how the eigenvector flow helps overcome weak principal curvature responses.

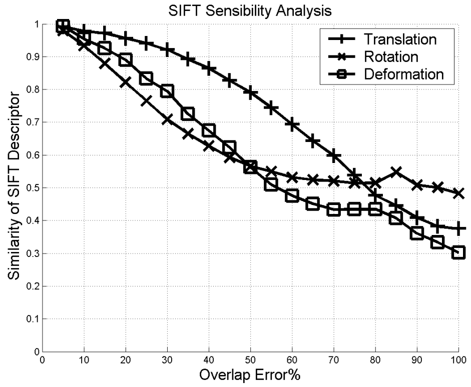


Figure 5. Sensitivity analysis of SIFT descriptor.

To determine a threshold value for the permitted amount of overlap, we analyze the sensitivity of the SIFT descriptor. We apply three transformations—translations from 1 to 10 pixels, rotations from 2 to 20 degrees and minor axis enlargements from 1 to 10 pixels—on all detected regions in the Inria dataset [19] and we compute the overlap errors and descriptor similarities between the original and transformed regions. As shown in Figure 5, descriptor similarity falls below 70% when the overlap error is greater than 30%. Thus, a region is "stable" to changes in scale if its overlap error is less than 30% when compared to another region in an adjacent scale. We keep all of these stable regions to maintain more descriptions for similar regions. Further, when the overlap error is less than 10% (producing a descriptor similarity above 90%), we only keep the region at the smaller scale and discard the other, almost identical, region.

4. Evaluation and Discussion

We evaluate PCBR in three ways: 1) qualitative visual inspection, 2) quantitative repeatability using a published framework [19], and 3) quantitative and qualitative evaluation using real world applications. Our Matlab implementation requires an average of 187.3 sec. to process a 2560×1920 image. We are porting our implementation to C/C++ where we expect a running time of less than 3 sec.

4.1. Visual Inspection

Figures 6 and 7 show PCBR detection results on a variety of image types. Fig. 6 shows PCBR detections on two graffiti images from the INRIA dataset [19] while Fig. 7 shows detection results (with background detections removed to improve visibility) for the face, motorbike, and car (rear) images from the Caltech dataset. From these images we note that PCBR detections appear to be evenly distributed, highly consistent, and robust to intra-class variations.

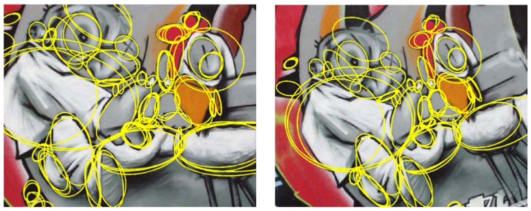


Figure 6. PCBR detections on the first and second graffiti images from the INRIA dataset [19].

Detectors	Motorbikes	Airplanes	Faces	Cars(Side)
PCBR	87.7	91.7	97.7	87
HL & HA	92.2	88.9	93.5	83

Table 1. Comparison of PCBR with the combination of Harris-Laplace and Harris-affine detectors on the Caltech dataset using Opelt's object-class recognition method [22].

4.2. Repeatability

Although the PCBR detector was designed for object-class recognition rather than repeatable wide-baseline feature matching, we still evaluate its repeatability and compare it to other detectors using the INRIA dataset and evaluation code [19]. Table 2 provides the average repeatability of the PCBR detector compared to various other detectors. Average repeatability is determined from the repeatability vs. transformation curves as output by the INRIA evaluation code (with the overlap error parameter set to 20%). As can be seen, PCBR is comparable to other detectors in terms of repeatability.

4.3. Applications

To quantitatively evaluate the PCBR detector on object recognition tasks, we have conducted two recognition tasks using the Caltech dataset and a database of larval stonefly images. To build a recognition system, we build SIFT descriptors for each detected (and normalized) region and then apply recent state-of-the-art object-class recognition algorithms for final classification. We then measure classification accuracy using the PCBR detector and compare it to the accuracy using other interest operators.

4.3.1 Object Recognition on Caltech Dataset

The Caltech dataset contains images from many different object classes. In this experiment, we measure recognition accuracy on four commonly-used object classes (motorbikes, airplanes, faces, and cars) using Opelt's object-class recognition method [22]. We use Opelt's Matlab code (adapted to use the PCBR detector) with all the default settings (see [22]) and with the only variation being that we use PCBR instead of the Harris-Laplace and Harris-affine

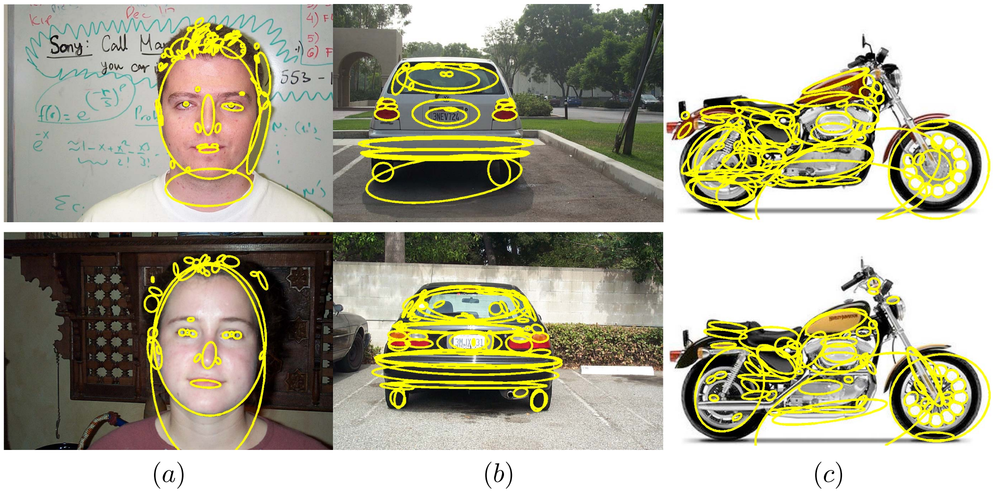


Figure 7. PCBR detections on faces, cars (rear), and motorbikes from the Caltech dataset.

Images	PCBR	Hessian-affine	Harris-Affine	MSER	IBR	EBR	Salient
Bikes	30.2	48.2	32.8	33.6	26.5	37.5	15.3
Trees	10.7	20.4	9.8	11.5	9.6	3.9	2.1
Boats	16.2	29.7	22.3	27.5	12.8	19.8	0.3
Leuven (cars)	37.6	40.0	32.0	66.7	34.4	30.1	17.6
Graffiti	35.5	17.7	13.0	51.7	19.7	16.7	2.1
Walls	16.6	24.5	17.3	31.4	14.7	11.1	0

Table 2. Average repeatability of PCBR and various other detectors with the overlap error parameter set to 20% on the INRIA dataset.

detectors. The recognition performance is evaluated using ROC equal error rates. Table 1 compares recognition accuracy using PCBR with those reported by Opelt in [22] (using Harris-Laplace and Harris-affine detectors). We see from Table 1 that the PCBR detector produces higher recognition accuracy than the combination of Harris detectors on three of the four object-class datasets. However, due to the already high recognition accuracy, none of the differences are statistically significant [6] (at a 95% level).

4.3.2 Object Recognition on Stonefly Dataset

Population counts of larval stoneflies inhabiting stream substrates are known to be a sensitive and robust indicator of stream health and water quality. Consequently, automated classification of stonefly larva can make great strides in overcoming current bottlenecks–such as the considerable time and technical expertise required–to large scale implementation of this important biomonitoring task. We thus evaluate the effectiveness of our PCBR detector on a more fine-grained object-class recognition problem, that of distinguishing between two related species of stonefly larva, *Calineuria californica* and *Doroneuria baumanni*. These two stonefly species are from the same taxonomic family and, as such, are very similar in appearance. Indeed, this problem is challenging even for humans and is akin to visually distinguishing between nearly identical car models. As such, this problem is more difficult than differentiating between faces and airplanes as per the Caltech dataset.

Figure 8 (a-b) shows four specimen images (and their relative sizes) from each of the two taxa. To verify the difficulty of discriminating these two taxa, we conducted an informal study to test the human classification accuracy of *Calineuria* and *Doroneuria*. A total of 26 students and faculty were trained on 50 randomly-selected images of *Calineuria* and *Doroneuria*, and were subsequently tested with another 50 images. Most of the subjects (21) had some prior entomological experience. The mean human classification accuracy is 78.6% correctly identified (std. dev. = 8.4).

We compare PCBR with the Kadir salient region detector and the Hessian-affine detector on the stonefly recognition problem. All classification settings are identical except for the detector. For this comparison, we use the Hessian-affine and salient region detectors available on the web. Figure 9 shows the detections for the four *Calinueria* images in Fig. 8(a). Notice again that the PCBR detections are well distributed and consistent.

We apply two state-of-the-art object-class recognition algorithms to the stonefly dataset: logistic model trees (LMT) by Landwehr *et al*. [10] and Opelt's method [22]. We use

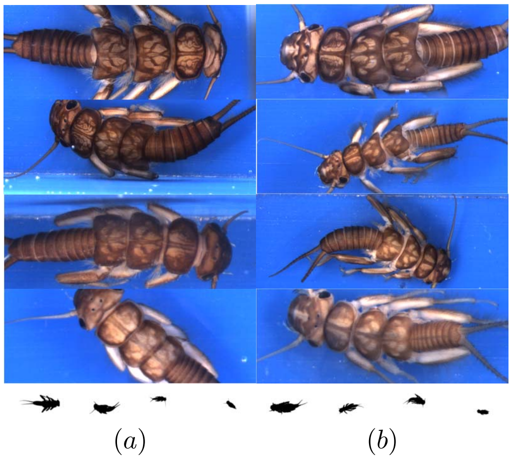


Figure 8. Visual comparison of *Calinueria* and *Doroneuria* and their relative specimen sizes. (a) Four different *Calinueria* and (b) *Doroneuria* specimens.

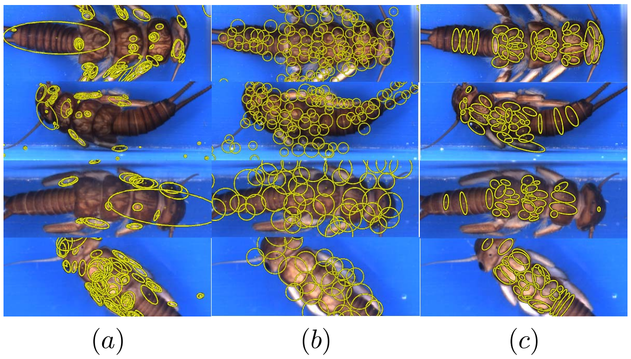


Figure 9. Comparison of three detectors on *Calinueria* images. (a) Hessian-affine, (b) Kadir salient regions, and (c) PCBR

Taxon	Specimens	Images
Calineuria	85	400
Doroneuria	91	463

Table 3. Specimens and images employed in the study.

our own LMT implementation and Opelt's Matlab code (adapted to use other detectors). The number of specimens and images used in this experiment is listed in Table 3 while Table 4 summarizes the classification accuracy for this two-class recognition problem. As can be seen, both classifiers yield better recognition accuracy with the PCBR detector than with the other two detectors. Further, the different recognition rates are statistically significant.

4.3.3 Symmetry Detection

Hessian Affine	Kadir Entropy	PCBR	Accuracy[%]	
			Opelt [22]	LMTs [10]
√			60.59	70.10
	√		62.63	70.34
		√	67.86	79.03

Table 4. *Calineuria* and *Doroneuria* classification rates comparison of different detectors when applied with Opelt's method and LMTs. A √ indicates that the corresponding detector is used.

Symmetry is common in biological and artificial objects. Since PCBR detects robust structure-based interest regions, it is good at detecting symmetrical regions in images containing objects with bilateral symmetry. To demonstrate this, we combine the PCBR detector with (our implementation of) the SIFT-based symmetry detection method of Loy and Eklundh [14] and test it on various images. Figure 10 shows the symmetrical detections is several images. We can see that the detected symmetry regions are quite accurate and distinctive, providing a valuable cue for the detection and recognition of symmetrical objects.

We also apply symmetry detection to choose good dorsal (i.e., back side) views of stonefly larvae from among the various poses. Dorsal views exhibit more bilateral symmetry than do other poses are helpful for classification due to the patterning on the specimens' backs. Figure 11 shows various poses of the stoneflies as contained in the database while Figure 12 shows the four selected dorsal views and their symmetrical PCBR detections. Based on visual inspection, the PCBR detector appears better suited for detecting the symmetric regions in the stonefly images than are other detectors.

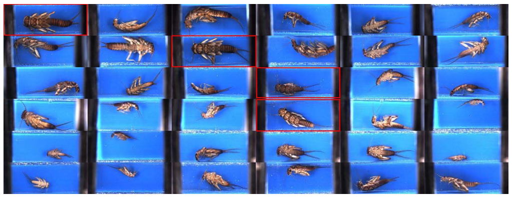


Figure 11. Different object poses in the stonefly database.

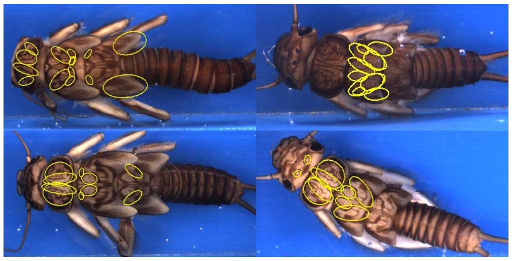


Figure 12. Good dorsal views selected using bilateral symmetry detection with PCBR.

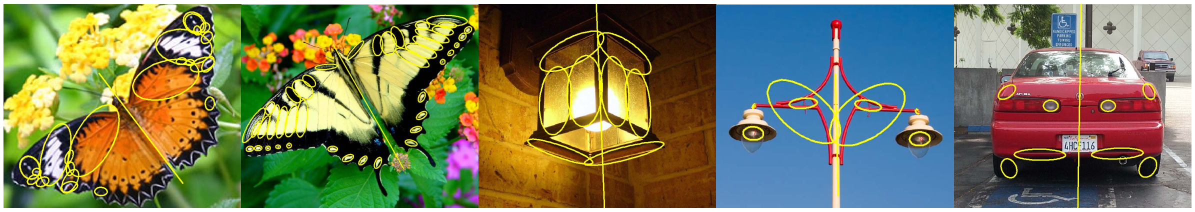


Figure 10. Bilateral symmetry detection using PCBR.

5. Conclusion and Future Work

This paper has presented a new structure-based interest region detector called Principal Curvature-Based Regions (PCBR) and has demonstrated its successful application to several tasks. The PCBR interest operator detects stable watershed regions within the multi-scale principal curvature image that describes both edge and curvilinear structures. Grayscale morphology and a new eigenvector-flow hysteresis thresholding provide for robust watershed detections. Further, PCBR achieves robust detections across multiple scales by selecting stable regions across consecutive scales. Finally, we have demonstrated the utility of our PCBR detector to reliable wide-baseline feature detection, object-class recognition, and symmetry detection.

Unlike detectors designed for wide-baseline matching of rigid static scenes, detectors for object recognition should be more intelligent in identifying object-class-relevant regions. As such, a future direction for this work is to develop discriminative interest operators and region descriptors that learn to detect and describe characteristic regions on a per object-class basis.

References

[1] H. Asada and M. Brady. The curvature primal sketch. *PAMI*, 8(1):2–14, 1986. 2

[2] A. Baumberg. Reliable feature matching across widely separated views. *CVPR*, pages 774–781, 2000. 2

[3] P. Beaudet. Rotationally invariant image operators. *ICPR*, pages 579–583, 1978. 2

[4] J. Canny. A computational approach to edge detection. *PAMI*, 8:679–698, 1986. 2

[5] R. Deriche and G. Giraudon. A computational approach for corner and vertex detection. *IJCV*, 10(2):101–124, 1992. 2

[6] T. G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1924, 1998. 6

[7] C. Harris and M. Stephens. A combined corner and edge detector. *Alvey Vision Conf.*, pages 147–151, 1988. 2, 3

[8] F. Jurie and C. Schmid. Scale-invariant shape features for recognition of object categories. *CVPR*, 2:90–96, 2004. 1, 2

[9] T. Kadir and M. Brady. Scale, saliency and image description. *IJCV*, 45(2):83–105, 2001. 2

[10] N. Landwehr, M. Hall, and E. Frank. Logistic model trees. *Machine Learning*, 59(1-2):161–205, 2005. 6, 7

[11] T. Lindeberg. Feature detection with automatic scale selection. *IJCV*, 30(2):79–116, 1998. 2

[12] T. Lindeberg and J. Garding. Shape-adapted smoothing in estimation of 3-d shape cues from affine deformations of local 2-d brightness structure. *Image and Vision Computing*, pages 415–434, 1997. 2

[13] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 2, 3

[14] G. Loy and J.-O. Eklundh. Detecting symmetry and symmetric constellations of features. *ECCV*, pages 508–521, 2006. 7

[15] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004. 2

[16] G. Medioni and Y. Yasumoto. Corner detection and curve representation using cubic b-splines. *CVGIP*, 39:267–278, 1987. 2

[17] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. *ECCV*, 1(1):128–142, 2002. 2

[18] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *IJCV*, 60(1):63–86, 2004. 2, 3

[19] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *IJCV*, 2005. 2, 3, 4, 5

[20] F. Mokhtarian and R. Suomela. Robust image corner detection through curvature scale space. *PAMI*, 20(12):1376–1381, 1998. 2

[21] H. Moravec. Towards automatic visual obstacle avoidance. *International Joint Conf. on Artificial Intelligence*, page 584, 1977. 2

[22] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer. Weak hypotheses and boosting for generic object detection and recognition. *ECCV*, pages 71–84, 2004. 5, 6, 7

[23] E. Shilat, M. Werman, and Y. Gdalyahu. Ridge's corner detection and correspondence. *CVPR*, pages 976–981, 1997. 2

[24] S. Smith and J. M. Brady. Susan–a new approach to low level image processing. *IJCV*, 23(1):45–78, 1997. 2

[25] C. Steger. An unbiased detector of curvilinear structures. *PAMI*, 20(2):113–125, 1998. 1, 3

[26] T. Tuytelaars and L. V. Gool. Wide baseline stereo matching based on local, affinely invariant regions. *BMVC*, pages 412–425, 2000. 2

[27] T. Tuytelaars and L. V. Gool. Matching widely separated views based on affine invariant regions. *IJCV*, 59(1):61–85, 2004. 1, 2