Reinforcement Matching Using Region Context

Hongli Deng[1] Eric N. Mortensen[1] Linda Shapiro[2] Thomas G. Dietterich[1]

[1]Electrical Engineering and Computer Science
Oregon State University
{deng, enm, tgd@eecs.oregonstate.edu}

[2]Computer Science and Engineering
University of Washington
shapiro@cs.washington.edu

Abstract

Local feature-based matching is robust to both clutter and occlusion. However, a primary shortcoming of local features is a deficiency of global information that can cause ambiguities in matching. Local features combined with global relationships convey much more information, but global spatial information is often not robust to occlusion and/or non-rigid transformations. This paper proposes a new framework for including global context information into local feature matching, while still maintaining robustness to occlusion, clutter, and non-rigid transformations. To generate global context information, we extend previous fixed-scale, circular-bin methods by using affine-invariant log-polar elliptical bins. Further, we employ a reinforcement matching scheme that provides greater robustness to occlusion and clutter than previous methods that non-discriminately compare accumulated bins values over the entire context. We also present a more robust method of calculating a feature's dominant orientation. We compare reinforcement matching to nearest neighbor matching without region context and to robust matching methods (RANSAC and PROSAC).

1. Introduction

Feature matching or correspondence is critical in many computer vision applications. Most feature matching tasks can be divided into one of three categories. The first application domain determines feature correspondences between multiple images of the same scene under different viewing conditions for tasks such as 3D reconstruction or recovering camera motion in a static scene. These applications usually need to recover epipolar geometry or solve for a rigid 3D motion model to find a consistent set of matching features. The second category involves object class recognition, where there is typically no rigid transformation to recover and the spatial relationships between features, as well as the descriptors identifying matching object "parts", can have considerable variation. The third application area is non-rigid object tracking in video. This domain combines the similar local appearance of matching features, as in the first domain, with non-rigid spatial geometry common in object class recognition.

For the first application domain, feature correspondence typically occurs by first matching local descriptors and then finding a set of consistent correspondences (or equivalently rejecting outliers) relative to some geometric constraints [2,17,18,19]. In the second and third application areas, the spatial arrangement of matched features can exhibit variations that are not accurately modeled with 3-D rigid transformations. Often, the features are organized into larger structures and matching is considered a global optimization problem. There are several possible approaches including graph-based models [6,7,10], fuzzy or relaxation algorithms [4,9], and spatial binning models [1,3,16].

Our method can be viewed as an extension to spatial binning. The spatial bins approach proposed by Belongie et al. [1] starts with a collection of shape points and builds, for each point, a histogram describing the relative distribution of the other points in log-polar space. Carneiro and Jepson [3] build log-polar bins around each feature and accumulate the weighted count of other features within each bin. Mortensen et al. [16] augment local descriptors to include global context information to develop a feature vector that includes both local features and global curvilinear information.

The above methods demonstrate that it is difficult to find an efficient matching method that is flexible enough to handle all kinds of correspondence tasks. Matching based on estimating a transformation matrix cannot handle non-rigid transformations-and the required planar assumptions are rarely satisfied in real-world images—while graph-based and optimization methods are often very computationally expensive. Our method can match regions strictly, as required by the first kind of application, and it can match regions with some deformations, as required by the second kind. Unlike the shape context method, which only analyzes the distribution of features, our method matches feature appearance as well. Both shape context and global context descriptors employ circular bins and thus cannot handle affine transformations. We use elliptical bins to make our method affine-invariant. Furthermore, in the bins in all previous spatial binning methods contains a single value that simply represents the accumulation of points, features, or pixel values within the bin, these methods are

susceptible to cluttered backgrounds and occlusion. We can achieve partial matching because our method is based on distributed local regions.

Our reinforcement matching has two advantages over sample consensus methods such as RANSAC [8], or more recently PROSAC [5]. First, reinforcement matching doesn't need a consistency model (e.g., epipolar geometric constraints) and consequently, our method works on any reasonable (including non-rigid/non-linear) transformation without requiring a constraint model, and consequently a sample set size. Second, when there are a high percentage of outliers, RANSAC is much less likely to select a sample set from among the inliers—which is necessary to compute the correct transformation. On the other hand, reinforcement matching is more tolerate by effectively ignoring outliers.

2. Reinforcement Matching

In this paper we use the Hessian-affine interest operator developed by Mikolajczyk and Schmid in [12, 15] due to its performance, repeatability and affine invariant properties. We use SIFT [11] to describe each detected region. Our reinforcement matching algorithm can be summarized as follows:

1) For each detected region, calculate the dominant gradient orientation and use it to choose the reference orientation of the ellipse region.
2) Scale the detected affine region (i.e., the innermost ellipse) to obtain two additional regions that are 8 and 16 times larger (the outer two ellipses in Fig. 1(a-b)). The features detected within these enlarged ellipses form the "region context" for the center feature.
3) Normalize the enlarged regions, including the positions of all the contained context features. Define context bins for each normalized region and construct, for each bin, a list of context features that fall in that bin.
4) Construct the initial matching distance matrix using Euclidean distance and local features only. From this matrix, a fixed fraction of one-to-one best matches are chosen to form "anchor regions".
5) Compute the final match score between each pair of regions by combining the Euclidean distance match score with the context score, which is computed by counting, for corresponding bins in the context of the two regions, the number of matching anchor regions they contain.

The details of our matching procedure are given below.

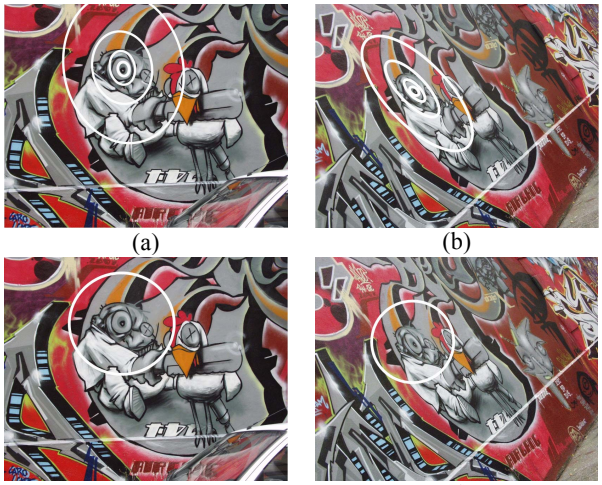


Figure 1: (a-b) Affine regions used to build the region context. (c-d) Circles shows that non-affine regions fail to capture similar context.

2.1. Building Region Context

To build the feature's region context for a detected feature, we enlarge the feature's original affine region while maintaining its elliptical shape. This methodology is based on the belief that the deformation of the area around the detected region is somewhat similar to the deformation of the center region. Figures 1(a-b) show sample of corresponding context regions for a pair of images. The innermost ellipse is the original detected region. The second one is used to calculate the SIFT descriptor. The outer two ellipses—which are eight and sixteen times larger than the inner ellipse—are used to build context bins. The size of the context bins follows the log-polar bin design of [1, 16]; thereby allowing for image deformations due to perspective and non-rigid transformation. Fig. 1 (c-d) shows that circular (non-affine) bins fail to cover the same area. We can see that the area enclosed by the circle in (c) is different from the area enclosed by the circle in (d) although the two circles have the same size relative to the initial keypoint's scale.

2.2. Dominant Orientation Calculation

A stable and robust reference orientation is critical to ensure rotation invariance for both the SIFT descriptor and the region context. Both Lowe [11] and Mikolajczyk [14] compute dominant gradient orientation in a small circular neighborhood around each keypoint. The size of the circular neighborhood is determined by the keypoint's scale but its shape is not affine-invariant. The gradient vector of every pixel in the circular region is used to build a histogram of gradient angles weighted by the gradient magnitude, and the orientation corresponding to the largest histogram bin is chosen as the dominant gradient.

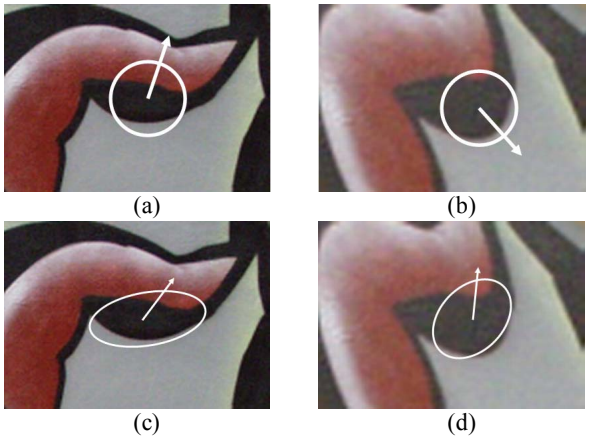


Figure 2: (a-b) Illustration of how the dominant orientation for a local feature can be affected by using circular neighborhoods that enclose different areas. (c-d) The dominant orientation computed using the affine detected region is more stable.

For images with uniform scale change, in-plane rotation, and even minor affine deformations, computing gradient orientation from a circular region is acceptable. However, using a circular region in the presence of large affine transformations does not produce a stable dominant orientation since the area contained within the circles will be different (see Figures 2(a-b)). On the other hand, calculating the dominant orientation using the elliptical regions is more stable since the enclosed areas more closely match (Fig. 2(c-d)).

To sample the gradient within an affine region, we use an efficient scan-line algorithm to determine the pixels contained within the ellipse. Centering the coordinate axis on the keypoint, the implicit equation of the ellipse is

$$Ax^2 + Bxy + Cy^2 = 1. \quad (1)$$

Given the eigenvalues (λ_1, λ_2) and eigenvectors (v_1, v_2) of the matrix

$$\boldsymbol{M} = \begin{bmatrix} A & B \\ B & C \end{bmatrix}, \quad (2)$$

the vertical scanline range of the ellipse is given by

$$y_{max} = \sqrt{A \cdot r_a^2 \cdot r_b^2}$$
$$y_{min} = -y_{max} \quad (3)$$

where

$$r_a = \frac{1}{\sqrt{\lambda_2}}, \quad r_b = \frac{1}{\sqrt{\lambda_1}} \quad (4)$$

are scale factors for the ellipse's major and minor axes, respectively.

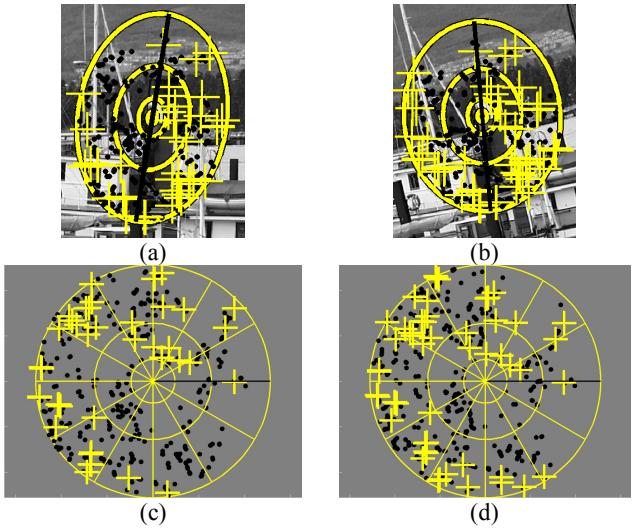


Figure 3: (a-b) Region context of two corresponding features. Yellow crosses are features within the top 10% of high confidence matched regions. Black dots are other detected features in this context. (c-d) Normalized region context of the two corresponding regions above.

For each horizontal scanline in the range $y_{min} \le y \le y_{max}$, the starting, x_{min}, and ending, x_{max}, points are obtained by solving the implicit ellipse equation (1) with known y value. Once the gradient values within the ellipse have been computed, the same histogram building process as in [11] is used to find the dominant gradient. In Section 3 we note that using the elliptical region to compute dominant orientation achieves better matching results than using the circular region (Fig.8).

2.3. Region Context and Reinforcement Matching

2.3.1. Choosing the Reference Orientation

After computing the dominant orientation, θ_D, we form a unit vector, $v_D = [\cos(\theta_D), \sin(\theta_D)]^T$, and use it to choose the orientation of the ellipse. Since the dominant orientation tends to point in the direction of the minor axis, v_1 or $-v_1$, we choose the reference orientation as the direction along the major axis, v_2 or $-v_2$, that produces a positive cross product with v_D. In other words, our reference orientation, α, is defined as

$$\alpha = \arctan\left(\mathrm{sgn}(v_2 \times v_D) \frac{v_{2,y}}{v_{2,x}} \right) \quad (5)$$

2.3.2. Region Context Selection

Given the equation for an ellipse in Eq. (1), a point (x, y) is within an ellipse that is S times larger if

$$Ax^2 + Bxy + Cy^2 \le S^2. \quad (6)$$

In Figure 3, the second innermost ellipse corresponds to

$S = 3$ and it is used to calculate the SIFT descriptor (as noted earlier). The third and fourth ellipses correspond to $S = 8$ and $S = 16$, respectively. Since the region inside the second ellipse is already described by the SIFT descriptor, the region context consists of all the features between the second ($S = 3$) and fourth ($S = 16$) ellipses. The black dots and yellow crosses in Fig. 3(a-b) are the features that fall within the region context. These features, of course, also represent an elliptical region with its own sizes and orientations, but are shown as crosses and dots to improve visibility.

2.3.3. Normalization of Region Context Bins

To ensure that each context feature maps to the correct context bin, we normalize the region context by using the ellipse parameters from the keypoint's second moment matrix. The transformation that maps the reference orientation to the x-axis and the inner ellipse ($S = 1$) to a unit circle is

$$\boldsymbol{M}' = \begin{bmatrix} \lambda_2 & 0 \\ 0 & \lambda_1 \end{bmatrix}^{-\frac{1}{2}} \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} = \begin{bmatrix} r_a & 0 \\ 0 & r_b \end{bmatrix} \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} \quad (7)$$

where λ_1, λ_2 are eigenvalues of Eq. (2), r_a, r_b are scale factors defined in Eq. (4), and α is the reference orientation from Eq. (5). The position of each context feature, $\boldsymbol{x}$, that falls within the region context is then mapped to its normalized position,

$$\boldsymbol{x}' = \boldsymbol{M}'\boldsymbol{x}\,, \quad (8)$$

and the context feature is added to the context bin that it falls in, as determined by the radial and angular position of $\boldsymbol{x}'$ in the normalized space. Rather than simply accumulate a count of the number of features in each bin, each context bin maintains a list of features (i.e., a list of SIFT descriptor indices). Given a feature, the feature's region context tells us what other features are near it and at what angle and distance. These context bin lists are the key to reinforcement matching since corresponding bins can be compared to determine the number of matching features in each bin while ignoring features that don't match.

2.4. Reinforcement Matching

The goal of reinforcement matching is to use the region context to efficiently improve matching accuracy by increasing the confidence of a good match between two features if they have a similar spatial arrangement of neighboring features. We first compute the $m \times n$ matching cost matrix that contains the Euclidean distance, $c(i, j)$ for $1 \le i \le m$, $1 \le j \le n$, between each pair of SIFT descriptors, where m is the number of features in the first image and n is the number in the second image. From these correspondences, we select a portion of the best matches (e.g., 20% of $\min(m, n)$) by iteratively selecting the best match in the matrix and then removing that match's row and column from further consideration. This process continues until reaching the target percentage. The selected matches are called anchor features. Note that this produces a one-to-one mapping. Figure 3 illustrates the two types of regions: anchor features (indicated with crosses) and other features (indicated with dots).

For each bin that has an anchor feature, we check whether the matching feature is in the corresponding bin of the other context and count the number of such feature matches. The final matching distance is

$$c'(i, j) = \frac{c(i, j)}{\log_{10}(10 + num_{support})} \quad (9)$$

where $num_{support}$ is the number of matched anchor features. If there are no matched anchor features in any of the context bins, then the denominator is unity and the central feature match is not reinforced. However, as the number of context matches increases, these matches reinforce the central match by increasing the denominator and thus lowering the final matching distance.

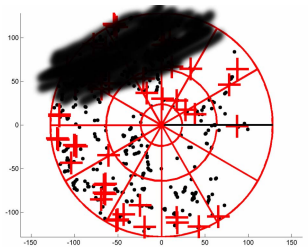


Figure 4: Illustration of how the region context is robust to occlusion. Reinforcement matching counts the number of matching features in corresponding bins. If a feature is occluded, it is simply ignored and features in other bins still provide sufficient support to reinforce the central feature match.

Figure 4 illustrates how this matching methodology is robust to occlusion and changes in background. If some of features are occluded in one or more bins, or if a context bin contains background that can change from one image to another, the missing features in those bins do not penalize the final matching distance (other than to reduce the support number) while other matches in other bins still contribute to sufficiently reinforce the central feature match. Note that this strategy provides a distinct advantage over global support methods that simply accumulate a single value in each bin. For example, if each bin simply summed up the number of feature/shape points [1] or gradient/curvature pixel values [16] in each bin, then bins that are occluded or contain differing background imagery would actually increase the matching

distance since the difference of accumulated bin values can be significant in these examples. Thus, using a single accumulated value in these cases can often lead to reduced matching rates.

3. Results

To evaluate performance, we use the INRIA dataset [13] that contains eight image sets representing five transformations (viewpoint change, zoom-rotation, image blur, JPEG compression, and lighting change). Each set contains six images at various degrees of transformation.

We compare our method with PROSAC (a recent, RANSAC-style robust matching method that uses progressive sample consensus) [5]. Since the INRIA image sets all represent homographies, they are well suited to RANSAC-style matching using epipolar geometric constraints. We use the same matching performance framework provided by Mikolajczyk and Schmid [15] (recall vs. 1 – precision curves) to evaluate matching performance for two different matching strategies: nearest neighbor (NN) and nearest-neighbor-ratio (NNR)—which finds the highest ratio of the nearest neighbor to the second nearest neighbor. The same experiments are done in all image sets. In every image set, images 2 through 6 are matched to the first image in their respective set. For each of these two matching strategies, we measure performance with (using $c'(i,j)$) and without (using $c(i,j)$) reinforcement matching and with PROSAC (using $c(i,j)$). Test results show that reinforcement matching provides higher accuracy than matching without region context on all images and is comparable to PROSAC with NN and better than PROSAC with NNR (Fig. 9).

One reason that reinforcement matching provides better matching rates than RANSAC methods is that, like shape context [1], our method provides for general-purpose two-dimensional constraints with some degree of positional flexibility (in that a reinforcing match can fall anywhere within a corresponding bin) while transformational constraints in RANSAC methods are typically more rigid and, in the case of epipolar geometry, only constrain matches to one-dimensional epipolar lines. However, Figure 5 demonstrates how highly textured images can still produce many duplicated patterns even along a one-dimensional epipolar line. Figure 9(f) shows the recall vs. 1 – precision curves for matching this image with the first image from this (the tree image) set.

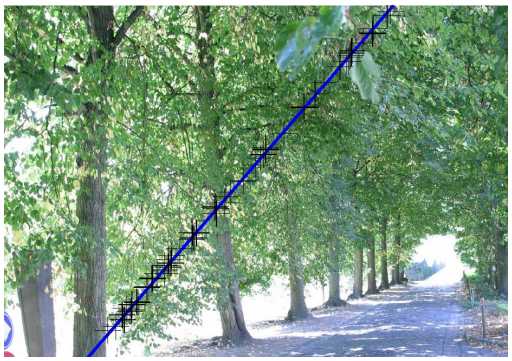


Figure 5: Example of how matching ambiguity can still exist even with 1-D epipolar constraints.

Another advantage of our method over RANSAC methods is that reinforcement matching doesn't need a transformation model and is therefore more flexible in that it can handle non-rigid or unknown transformations. We demonstrate this flexibility by matching images that have undergone affine, projective, polynomial, piecewise linear, sinusoidal and barrel transformations (some of which are shown in Figure 6). All seven transformations are applied to every image in the INRIA data set and compared with the first, untransformed image from each corresponding set. Results show that our method can increase the matching rate 8% on average over matching without region context (Figure 7). We do not show results using RANSAC or PROSAC since an epipolar model is clearly incorrect and, consequently, these methods typically fail to arrive at a correct consensus. While we could apply the correct transformation, since it is known, a different consistency model would have to be applied for each of the seven transformations. On the other hand, reinforcement matching does not require a transformation model and, as such, can be applied directly to all of the images regardless of the transformation.

To evaluate the performance of our new dominant gradient calculation, we compared the new method with the standard method on all images in the INRIA dataset. On images without large affine changes, matching performance using our new method is the same or slightly better than that of the previous method. For images with large affine changes, the performance of our method is noticeably better (Fig. 8).

To evaluate the influence of the number of bins, we measured performance using configurations with 8, 16 and 24 bins. The configuration with 24 bins provides the best performance, but the difference between 24 bins and 16 bins is marginal.

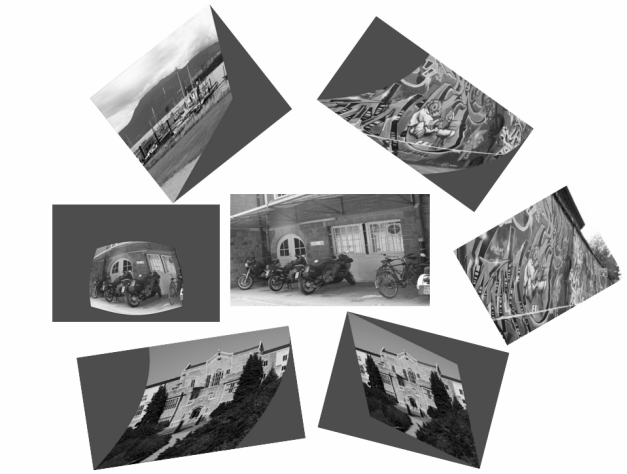


Figure 6: Examples of transformed images from the INRIA data.

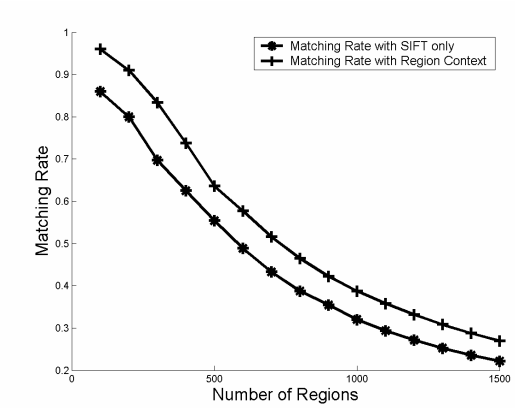


Figure 7: Matching rate on changing viewpoint images of a structured scene.

We examined the effect of using 5%, 10%, 20%, 40%, 60% and 100% of the total matched features as anchors. Variations of the recall score can be as large as 10%. The basic trend is that a higher percentage of anchor features improves performance— however, we achieve a rate of diminishing returns at about 20%. An exception to this increasing trend is that on some images with large zoom and rotation, the best-matched features have many errors, resulting in decreased performance with increased percentage of anchor features.

The worst case computational complexity for n features is $O(n^3)$, but this only occurs when all the anchor features are in a single bin for all the matches. In practice, the complexity is $O(n^2m)$ where $m << n$ is the average number of anchor points in a bin.

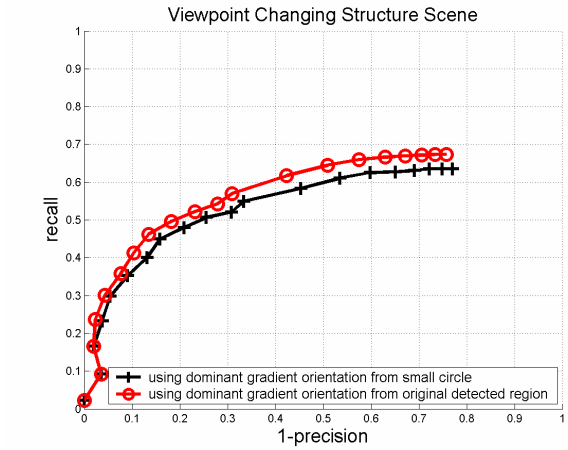


Figure 8: Comparison of two methods for dominant orientation calculation.

4. Conclusion and Future Work

This paper has presented a method including global context information into local feature matching, while still maintaining robustness to occlusion, clutter, and non-rigid transformations. The log-polar context bins employ a matching scheme that reinforces a local match with spatially consistent neighboring matches. Reinforcement matching also provides greater robustness to occlusion and clutter than previous methods that non-discriminately compare accumulated bins values over the entire context. Our evaluation indicates that matching with region context increases matching performance compared to matching without region context. Reinforcement matching is more robust and flexible than RANSAC-style methods in that it effectively ignores outlier matches without requiring a consistency model for each type of transformation. This paper also describes a more robust method of calculating a feature's dominant orientation.

Capturing spatial deformations with log-polar bins simplifies the description of spatial relations, but it still fails on large deformations. Also, features that are near the border of a bin can sometimes fall into the wrong bin and fail to be correctly matched. Future research will explore methods that relax these constraints and achieve a more flexible matching methodology.

Acknowledgements

The authors gratefully acknowledge the support of the NSF under grant IIS-0326052.

References

[1] S. Belongie, J. Malik, and J. Puzicha, "Shape Context: A New Cescriptor for Shape Matching and Object Recognition," in *NIPS*, pp. 831-837, 2000.

[2] M. Brown and D. G. Lowe, "Invariant Features from Interest Point Groups," in *BMVC*, pp. 656-665. 2002

[3] G. Carneiro and A. Jepson, "Pruning Local Feature Correspondences using Shape Context," in *ICPR*, Vol. 3, pp. 16-19, 2004.

[4] H. Chui and A. Rangarajan, "A New Algorithm for Non-Rigid Point Matching," in *CVPR*, pp. 44-51 2000.

[5] O. Chum and J. Matas, "Matching with PROSAC-Progressive Sample Consensus," in *CVPR,* Vol. I, pp. 220-226, 2005.

[6] D. Crandall, P. Felzenszwalb and D. Huttenlocher, "Spatial Priors for Part-Based Recognition using Statistical Models," in *CVPR*, pp. 10-17, 2005

[7] A. D. Cross and E. R. Hancock, "Graph Matching with a Dual-Step EM Algorithm," *PAMI*, pp. 1236-1253, 1998

[8] M. A. Fischler, R. C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Comm. of the ACM*, Vol. 24, pp 381-395, 1981.

[9] G. E. Hinton, C. K. Williams, M. Revow, "Adaptive Elastic Models for Hand-Printed Character Recognition," in *NIPS*, pp. 512-519, 1992.

[10] G. Lohmann and D. Y. von Cramon, "Automatic Labeling of the Human Cortical Surface using Sulcal Basins," *Medical Image Analysis*, **4**, pp. 179-188, 2000.

[11] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *IJCV*, **60**(2), pp. 91-110, 2004.

[12] K. Mikolajczyk and C. Schmid, "An Affine Invariant Interest Point Detector," in *ECCV*, Vol. I, pp. 128-142, 2002.

[13] K. Mikolajczyk and C. Schmid, "Scale & Affine Invariant Interest Point Detectors," *IJCV*, **60**(1), pp. 63-86, 2004.

[14] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *PAMI*, **27**(10), pp. 1615-1630, 2004.

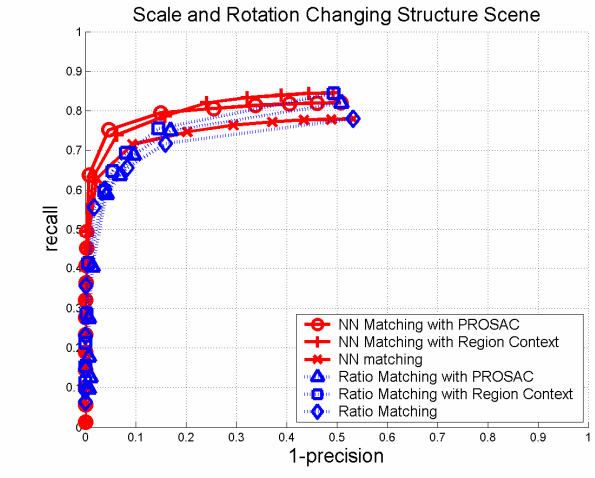
[15] K. Mikolajczyk, et al., "A Comparison of Affine Region Detectors," accepted to *IJCV*, 2005.

[16] E. Mortensen, H. Deng and L. Shapiro, "A SIFT Descriptor with Global Context," in *CVPR*, Vol. I, pp. 184-190, 2005.

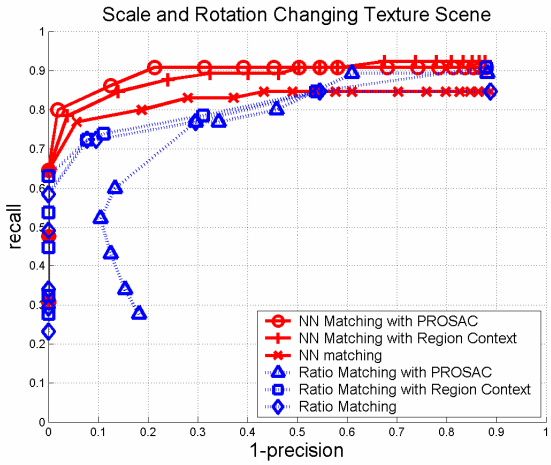
[17] P. Pritchett and A. Zisserman, "Wide Baseline Stereo Matching," in *ICCV*, pp. 754-760, 1998.

[18] D. Tell and S. Carlsson, "Wide Baseline Point Matching using Affine Invariants Computed from Intensity Profiles," in *ECCV*, pp. 814-828, 2000

[19] Z. Zhang, R. Deriche, O. Faugeras and Q.-T. Luong, "A Robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geometry," *Artificial Intelligence Journal*, **78**, pp. 87-119, 1995.



(a)



(b)

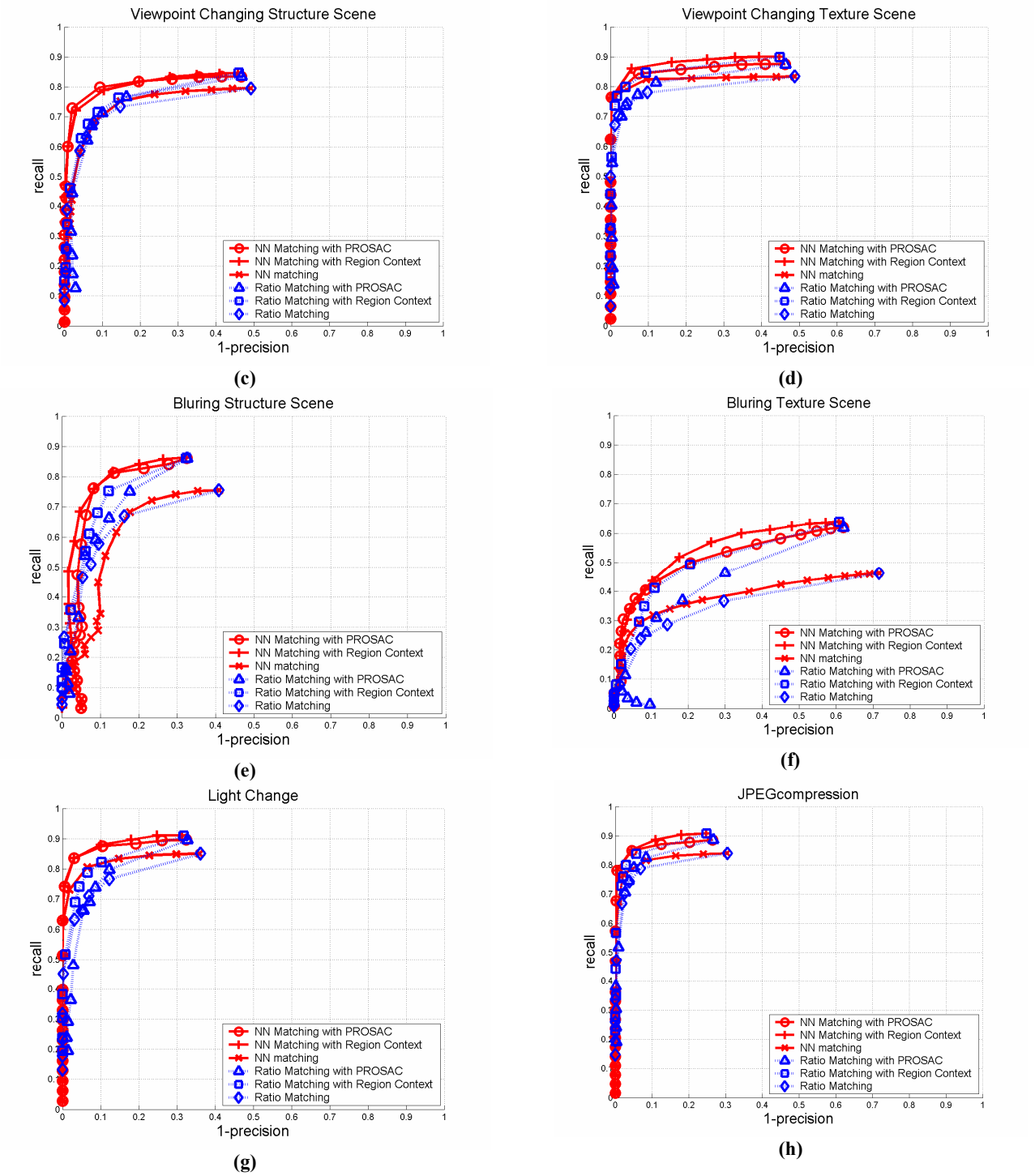


Figure 9: Comparison of matching performance with and without region context and with PROSAC for two matching strategies using six types of image transformations: (a) boat (previous page), (b) bark (previous page), (c) graffiti, (d) wall, (e) bike, (f) trees, (g) Leuven, (h) UBC. Images can be downloaded from: http://www.robots.ox.ac.uk/~vgg/research/affine/index.html.