To appear in CVPR 2005.

Coherent Regions for Concise and Stable Image Description

Jason J. Corso and Gregory D. Hager
Computational Interaction and Robotics Lab
The Johns Hopkins University
Baltimore, MD 21218
{jcorso|hager}@cs.jhu.edu

Abstract

We present a new method for summarizing images for the purposes of matching and registration. We take the point of view that large, coherent *regions in the image provide a concise and stable basis for image description. We develop a new algorithm for image segmentation that operates on several* projections *(feature spaces) of the image, using kernel-based optimization techniques to locate local extrema of a continuous scale-space of image regions. Descriptors of these image regions and their relative geometry then form the basis of an image description.*

We present experimental results of these methods applied to the problem of image retrieval. On a moderate sized database, we find that our method performs comparably to two published techniques: Blobworld and SIFT features. However, compared to these techniques two significant advantages of our method are its 1) stability under large changes in the images and 2) its representational efficiency. As a result we argue our proposed method will scale well with larger image sets.

1. Introduction

In this paper, we consider the problem of matching (or registering) differing views of a scene to each other. This, problem, which has received an immense amount of attention over the last decade, is currently solved using two different approaches.

One set of approaches, pioneered by Schmid and Mohr [21] and extended in many recent papers [9, 12, 8, 7, 15, 16, 22], makes use of local region descriptors for indexing and matching. The general idea of such approaches is to locate regions of high texture content using an interest operator and to then create indices for matching. The key to good performance is to create interest operators and match indices that are insensitive to geometric and photometric image distortions. The advantage of the approach is generally the robustness of matching to occlusion, changes in lighting, and moderate changes of viewpoint. The disadvantages are the need to identify such local image regions and (typically) the use of only gray-scale image projections. In particular, large areas of the image are potentially discarded as "untextured" and therefore unusable by the method. Mikolajczyk and Schmid [17] evaluated the performance of several local image descriptors. Their evaluation tested the descriptors' stability to rotation, scaling, affine transformations, and illumination changes. The study showed that SIFT [12] features performed the best over all conditions.

Another set of approaches, exemplified by Malik et al. [13, 1] and Greenspan et al. [6], instead represents images through segmentation. This approach is particularly appealing for image retrieval problems where the goal is to find similar, rather than exactly matching, images. The advantage is that large areas of the image tend to be stable across large changes in viewpoint and can be matched in a spatially approximate manner. The disadvantages are that it is necessary to have an efficient yet stable segmentation process and to find image cues that are themselves stable over variations in pose and lighting.

In our work, we consider a "middle ground." Namely, our goal is to create interest operators that focus on homogeneous regions, and local image descriptors for these regions. To this end, we perform a sparse image segmentation and then index images based on the results of that segmentation. The segmentation is performed in parallel on several scalar image projections (feature spaces) using kernel-based optimization methods. The optimization evaluates both the size (large regions tend to have high stability across widely disparate views) and the coherency (e.g. similar color, texture, depth, or image gradient) of region content. Once a region is located, its description is composed of simple kernel-weighted means of the coherent content. This description is concise: it is stable under drastic changes in viewpoint, and it is insensitive to photometric changes provided the initial image projections are. In particular, the kernel-based optimization can easily be made fully invariant to affine geometric deformations. Finally, since we compute multiple image regions, images can be geometrically registered in a manner similar to interest point-based registration.

In principle, our method is most similar to Schaffalitzky and Zisserman [20]. They use the texton segmentation [1] and create a texture region descriptor that is invariant to affine geometric and photometric transformations. They robustly estimate the epipolar geometry of a wide baseline system by matching these regions. While emphasis on scene-retrieval and registration based on regions as opposed to points is similar to their work, we differ in the region detection and description. The remainder of this paper details our kernel-based segmentation methods and provides preliminary comparative experimental results suggesting region-based matching performs comparably with other published image matching methods.

2. Coherent Region Clustering

Scale is a crucial parameter in the analysis of objects in images. In our case, there are two essential notions of scale: the scale of the image content (e.g. texture or edges), and the scale of an associated spatial kernel function used to summarize image content. In both cases, there is no universally accepted method for choosing an optimal scale. Lindeberg proposed a set of scale selection principles [11] for feature detection and image matching, and a technique [10] for building a gray-level blob and scale-space blob representation of an image. Comaniciu et al. [5] proposed the variable bandwidth mean shift to solve this problem (in the context of kernel-based density estimation [4]). Collins [2] applied Lindeberg's general scale selection principles [11] to extend the kernel-based mean shift tracking to refine the scale of the object being tracked. Okada et al. [19] presented a method for the creation of an anisotropic, Gaussian scale-space by extending Lindeberg's [11] isotropic scale-space methods.

In our work, we focus primarily on determining the correct scale of a spatial kernel for summarizing image content. Let an image $\mathbf{I} \doteq \{\mathcal{I}, I\}$ be a finite set of pixel locations $\mathcal{I}$ (points in $\mathbb{R}^2$) together with a map $I : \mathcal{I} \rightarrow \mathcal{X}$, where $\mathcal{X}$ is some arbitrary value space of dimension d. The image band j at pixel location i is denoted $I_j(i)$. Thus, for our purposes the *image* is any scalar or vector field: a simple grayscale image, an YUV color image, a disparity map, a texture-filtered image, or any combination thereof.

A *coherent* region in an image is a connected set of (relatively) homogeneous pixels. We describe a coherent region $\boldsymbol{\theta}$ as a two-dimensional Gaussian weighting function, or *kernel*, K with 4 parameters,[1] two spatial locations and two corresponding scale parameters $\boldsymbol{\theta} = \{\mu_x, \mu_y, \sigma_x, \sigma_y\}$. Thus, for a pixel location $\mathbf{x} = (x, y)^\mathsf{T} \in \mathcal{I}$, the kernel is written

$$K(\boldsymbol{\theta}, \mathbf{x}) \doteq \frac{e^{-\frac{1}{2}\left(\frac{(x-\mu_x)^2}{\sigma_x^2} + \frac{(y-\mu_y)^2}{\sigma_y^2}\right)}}{Z} \tag{1}$$

where Z is a normalizing scalar.

2.1. Scalar Projections

Kernels are applied to scalar projections of the image. The intuition is that various projection functions will map a region of consistent image content to a homogeneous image patch in the scalar field: for example, there is some texture and/or color projection function such that an image of a plaid shirt will be mapped to a relatively homogeneous scalar field. Thus, by choosing appropriate scalar projections, we can capture a variety of different coherent image content.

To that end, define a function $P : \mathcal{X} \rightarrow \mathbb{R}$ that *projects* regions of the d-dimensional image onto a one-dimensional scalar field. Essentially, each projection is defining a new *feature space* in which to analyze the input image. If we assume an image where $d = 3$ and $\mathcal{X} =$ RGB, then a feasible projection is a simple linear combination with coefficients $\{c_r, c_g, c_b\}$ of the pixel color components [3]:

$$S(i) = c_r I_r(i) + c_g I_g(i) + c_b I_b(i), \quad \forall i \in \mathcal{I}. \tag{2}$$

Other potential projections include neighborhood variance, periodicity of the image, dominant gradient direction, and so forth. It is even plausible to create a specific template matching projection; for example, if we want to find a certain road-sign in the image, then the projection is simply computed by convolving the image by a road-sign template. The methodology we propose is general and the construction of these projections is application dependent. In this paper we give a simple set of projections (§ 5). Additionally, the projections affect the invariance properties of the image description. We defer such a discussion to § 3.5.

2.2. Region Statistics

We define the statistics that will be used in region detection and description. For projection P,

$$\text{Mean}_P(\boldsymbol{\theta}, \mathbf{I}) = \sum_{\mathbf{x} \in \mathcal{I}} K(\boldsymbol{\theta}, \mathbf{x}) P(I(\mathbf{x})) \tag{3}$$

$$\text{Var}_P(\boldsymbol{\theta}, \mathbf{I}) = \sum_{\mathbf{x} \in \mathcal{I}} K(\boldsymbol{\theta}, \mathbf{x}) P(I(\mathbf{x}))^2 - \text{Mean}_P(\boldsymbol{\theta}, \mathbf{I})^2. \tag{4}$$

We use the kernel-weighted mean and variance instead of uniformly weighted statistics because the pixels in the center of the region are more likely to have stronger coherency than the pixels on the outer parts of the region. We show an experiment to justify this claim in Fig. 8.

[1] For the moment, we disregard rotations of anisotropic kernels and image skew which would be necessary to ensure full affine invariance of the kernel-based segmentation.

3. Region Detection and Refinement

3.1. Initial Seed Detection

Marr and Hildreth [14] first proposed the use of the Laplacian of a Gaussian (LoG) for distinguishing homogeneous regions from the drastic changes in intensity that separate them. More recently, Lowe [12], among others [9], used a Difference of a Gaussian (DoG) to approximate the LoG filter. They construct a dense, discrete scale-space of DoG responses and then perform an explicit search for stable points (local extrema in space and scale).

To detect seed points, we create a coarse, discrete scale-space of isotropic DoG responses by sampling a few (in our experiments just 2) large scales. This coarse sampling is sufficient for seed detection because we later refine each candidate seed and localize it in both space and scale. Similar to Lowe, we look for local extrema in the DoG response to detect seeds. However, since we are coarsely sampling scale-space, we analyze each 2D DoG-response separately (Lowe searches for extrema in 3D scale-space).

We define a seed with three parameters $\hat{\boldsymbol{\theta}} = \{\mu_x, \mu_y, \sigma\}$ where μ_x, μ_y are the spatial coordinates and σ is an isotropic scale. We set the scale of the seed to one-third of the scale of the LoG filter. Intuitively, this one-third scale factor shrinks the kernel to the homogeneous region at the filter's center. In contrast, Lowe scales the region by a factor of 1.5 because the SIFT keys function best in regions of high variance (the region including its surrounding areas, for example).

3.2. Refining the Seeds into Regions

In the second stage of processing, we take the detected seeds and independently refine their spatial location and (anisotropic) scale with respect to the original image. Thus, we create a continuous scale-space of regions.

The objective function we optimize consists of two competing terms: a homogeneity term and a scale term:

$$O_P(\boldsymbol{\theta}, \mathbf{I}) = \frac{\text{Var}_P(\boldsymbol{\theta}, \mathbf{I})}{\text{Mean}_P(\boldsymbol{\theta}, \mathbf{I})^2} + \frac{\tau}{\sqrt{\sigma_x \sigma_y}}. \qquad (5)$$

The first term measures the variance of the kernel-weighted region normalized by the squared mean. Thus, for poorly projected, inhomogeneous regions, the first term will be high. The second term is an ad-hoc penalty that prefers large regions. τ is a tuning parameter; in all of our experiments, we choose $\tau = 1$. In Fig. 1, we show the function response of (5) for a synthetic image containing a single homogeneous region in a field of noise.

The minima of (5) correspond to large, coherent regions. Thus, we refine the seeds from § 3.1 by minimizing the function in (5):

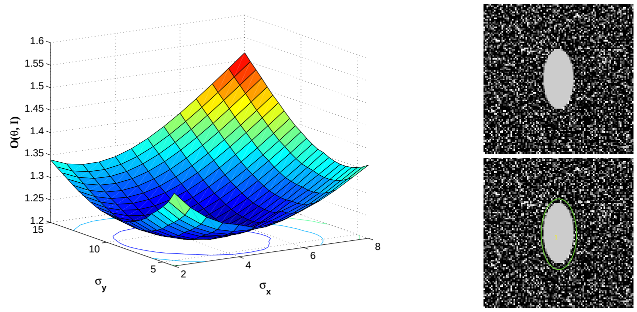

Figure 1: (left) (5) response for varying σ along each dimension for the synthetic image on the top-right. Spatial location is fixed at the (known) region center. (bottom-right) Extracted ellipse overlayed on the image.

$$\begin{aligned}\boldsymbol{\theta}^* &= \arg\min_{\boldsymbol{\theta}} \left[\frac{\text{Var}_P(\boldsymbol{\theta}, \mathbf{I})}{\text{Mean}_P(\boldsymbol{\theta}, \mathbf{I})^2} + \frac{\tau}{\sqrt{\sigma_x \sigma_y}}\right] \\ &= \arg\min_{\boldsymbol{\theta}} \left[\frac{\sum_{\mathbf{x}\in\mathcal{I}} K(\boldsymbol{\theta}, \mathbf{x}) P(I(\mathbf{x}))^2}{\left[\sum_{\mathbf{x}\in\mathcal{I}} K(\boldsymbol{\theta}, \mathbf{x}) P(I(\mathbf{x}))\right]^2} + \frac{\tau}{\sqrt{\sigma_x \sigma_y}}\right]. \end{aligned} \qquad (6)$$

This objective function can now be optimized using classical descent methods. We have experimented with both gradient-descent and second-order Newton-style minimization algorithms and have found both to provide satisfactory results. As we would expect, the second-order methods converge more quickly but are more sensitive to initialization. During minimization, we must ensure the scale parameters remain positive and the spatial parameters remain on the image lattice; we set explicit boundaries on scale ($[2, 60]$ pixels) and location. If these bounds are violated, we terminate optimization (removing the region).

It is worth noting that, in order to implement this optimization efficiently, several algorithmic optimizations are possible. In particular, we precompute kernels for a specific set of scales and resample images dynamically to the appropriate scale. With these optimizations, a complete optimization for one region consumes about 0.1 sec.

3.3. Merging and Annotation

Let $\mathcal{B}$ denote the set of active regions in an image. For a region $B \in \mathcal{B}$, denote the parameters by $\boldsymbol{\theta}(B)$. Since multiple seeds may converge to the same minima of (5), we perform a simple region merging procedure. Define the distance[2] between a pair of regions $B, C \in \mathcal{B}$ as,

$$d(B, C) = \|\boldsymbol{\theta}(B) - \boldsymbol{\theta}(C)\|_2. \qquad (7)$$

[2] We have experimented with more mathematically justified distances (e.g. Kullback-Leibler Divergence), but found them to have little or no effect on the merging process because, here, we are only merging regions that are near equal.

Fix a threshold α and define an empty set of merged regions $\hat{\mathcal{B}} = \emptyset$. Then, for each pair of regions $B, C \in \mathcal{B}$ solve

$$\hat{\mathcal{B}} = \hat{\mathcal{B}} \bigcup \left\{ \begin{array}{ll} B & d(B,C) < \alpha \\ \{B,C\} & \text{otherwise} \end{array} \right. . \qquad (8)$$

Although this is of quadratic order, we have found the number of regions was significantly reduced (about 25% on average) after the merging procedure. This reduction is insensitive to the threshold chosen.

Then, given a set of projections $\mathcal{P}$, for each region $B \in \hat{\mathcal{B}}$ annotate it by computing the kernel-weighted mean under each projection:

$$B_p = \text{Mean}_p(\boldsymbol{\theta}(B), \mathbf{I}), \quad \forall p \in \mathcal{P}. \qquad (9)$$

The resulting image summarization can be interpreted as a Gaussian mixture model over the joint feature-spatial space (with infinitesimal variance in the feature spaces). In Fig. 2, we show an example image with its representative regions under the three (red, green, and blue) color projections.

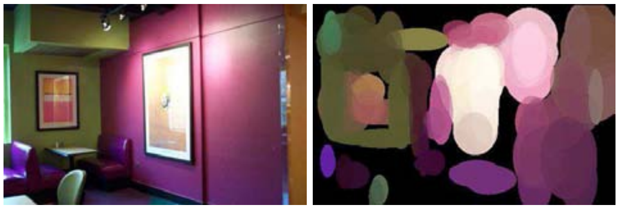

Figure 2: An image from the dataset and its representative color regions extracted with our technique.

3.4. Algorithm Summary

In this section, we summarize the complete algorithm which uses the local minima of a continuous scale-space as representative coherent regions in the image description. For a given input image $\mathbf{I}$, define a set of projections $\mathcal{P}$ and an initial, empty set of regions $\mathcal{B}$ and carry out the following steps:

1. Under each projection independently,
   - (a) Detect seeds. (§ 3.1).
   - (b) Independently, minimize the function in (6) to refine each seed.
   - (c) Add convergent regions to $\mathcal{B}$.
2. Merge $\mathcal{B}$ (8).
3. Annotate remaining regions in $\mathcal{B}$.

3.5. Properties

The coherent regions we present in this paper have a number of good properties: **stability/invariance**, **conciseness**, and **scalability**. The region description is implicitly invariant to rotation and translation in the image because it is simply a vector of kernel-weighted means. Since the image description is composed of a number of independent regions, like other local descriptor methods [21], it is robust to occlusion. In addition, using the kernel functions to weight the region statistics increases the robustness since it weighs pixels based on their distance from the region center.

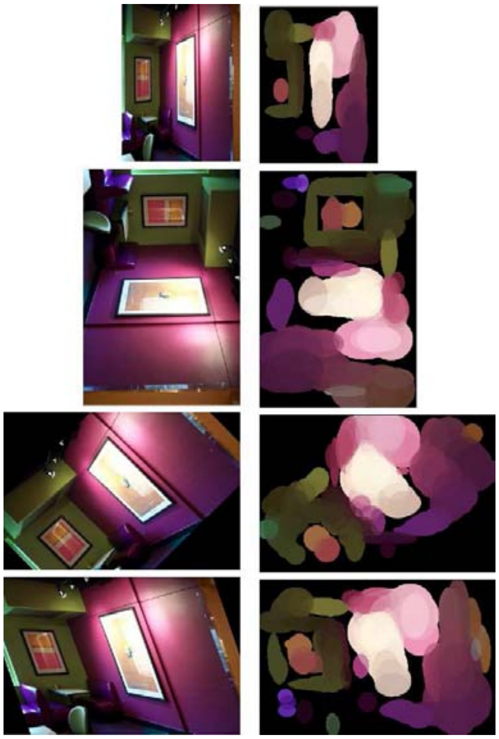

Figure 3: The coherent regions extracted are robust to affine distortion of the image even though we currently do not include kernel rotation which would be required for full affine invariance (each row is a pair, see text for explanation).

We claim that our method is robust to affine distortions in the image. In Fig. 3 we show the extracted regions (using the RGB projections for exposition) for the same image as Fig. 2 after it has been transformed by different affine maps: (row-wise) halving the aspect ratio, $90°$ rotation, $45°$ rotation, and a shear. From the figure, we see that roughly the same regions are extracted. It is important to note that for each extracted region, the kernel-weighted mean is stored, and thus, the precise geometric parameters of the regions may be slightly different. We include an experiment in § 5 (Fig. 6) in which we query the database with images that have been distorted. The experiment finds the affine distortion effects a minor change in the precision-recall for our technique while causing a large change for the SIFT method.

These invariance properties are dependent on the specific scalar projections employed. For instance, if the scalar projection is designed to extract *vertical* texture (*y-gradient* in the image), then the region's description under this projection is no longer rotationally invariant or robust to affine distortion. The projections we use in this paper all produce rotationally invariant and affinely robust descriptions ($\S$ 5). We are currently pursuing a more concrete definition for the constraints on the projections to ensure these invariance/robustness properties and leave it for future work.

It is clear that the image description is concise because we store simply a small set of kernel-weighted means per region. Thus, the storage requirement for our technique will not prohibit its scaling to larger databases.

4. Matching

Earlier, we mentioned that we consider the problem of matching differing views of a scene to each other. In this section, we discuss our approach for using the coherent region-based image description to address this problem.

Given a pair of images $\mathbf{I}^1, \mathbf{I}^2$ and their corresponding region sets $\mathcal{B}^1, \mathcal{B}^2$ computed from the same set of projections $\mathcal{P}$, the problem of matching can be approached on two levels: qualitative and quantitative matching. In qualitative matching, we address image content similarity; i.e. based on the two region sets, how similar are images $\mathbf{I}^1$ and $\mathbf{I}^2$? In quantitative matching, we quantify how much spatial coherence exists between the two images? Spatial coherence, in this case, is defined as the pixel-area in each image where matching regions overlap. We can then, for instance, maximize the amount of overlap region to compute the parameters of a geometric transformation relating the two images. In this paper, we focus on qualitative matching, and we use the same approach for all three methods in our comparative analysis.

To compute image similarity, we simply compute the sum-of-squared distance between the two region feature vectors; that is, for any two regions a, b:

$$s(a, b) = \sum_{p \in \mathcal{P}} (a_p - b_p)^2 \tag{10}$$

Thus, for each region $B^i \in \mathcal{B}^1$, we find its nearest neighbor $B^* \in \mathcal{B}^2$:

$$B^* = \arg\min_{B^j \in \mathcal{B}^2} s(B^i, B^j). \tag{11}$$

We repeat this procedure for each region in $\mathcal{B}^2$ and keep only those matches which are consistent in both directions.

5. Experiments in Image Retrieval

In this section, we discuss the techniques proposed in this paper for the task of image retrieval. For these experiments, we use a moderate sized dataset of 48 images[‡] taken of an indoor scene from widely varying viewpoints and with drastic photometric variability (a subset of the dataset is shown in Fig. 4). We hand-labeled the dataset; two images are said to be *matching* if there is any area of overlap between them. We can see from the images that the number of matches for each image has a high variation.

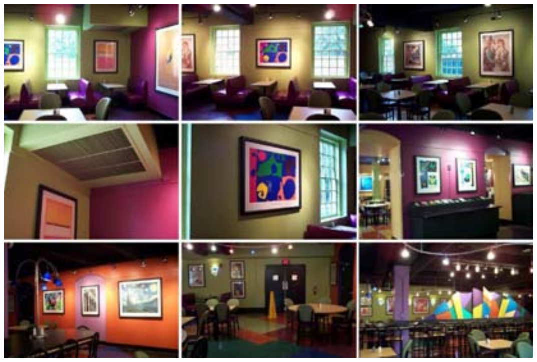

Figure 4: A subset of the dataset (chosen arbitrarily) used in the retrieval experiments.

Denote the three bands of the input image as R, G, B and S as their gray projection. Unless otherwise noted, we use a set of 5 projections: the 3 opponent color axes $((R+G+B)/3$,$(R-B)/3$, and $(2G-R-B)/4)$ which were experimentally shown by [18] to perform well in color segmentation, and 2 projections that measure neighborhood variance in S with window sizes of 16 and 32 pixels. As noted earlier, many other more sophisticated methods are plausible to capture texture information in the image projections; we leave such experimentation to future work.

We use the standard precision-recall graphs to present the matching results. The precision is defined as the fraction of true-positive matches from the total number retrieved and the recall is the fraction of matching images that are retrieved from the total number of possible matches in the database. First, we compare our technique to two techniques in the literature: SIFT Keys [12] and Blobworld [1].

SIFT is an example of a local, affine-insensitive and scale-invariant interest point descriptor. For matching, we use the same nearest-neighbor scheme as discussed in $\S$ 4. Note that additional geometric constraints are plausible for both our method and SIFT Key matching, but we do not employ any of them in order to keep the comparisons between methods fair. Blobworld is an example of using segmented image regions as the description. To measure matches using their provided source code, we used blob-to-blob queries.

[‡] The complete dataset can be found on the www at `http://www.cs.jhu.edu/~jcorso/r/regions/`.

For a query image $\mathbf{I}^1$ with regions $r_1, \ldots r_n$, we queried the database independently for each region r_i and maintained accumulators for each image. The final matches for the query image were those images with the highest accumulators after queries for all n regions had been issued.

	Average Number of Elements	Size[†] per Element (in Words)	Average Size (in Words)
Our Technique	159	5	797
Blobworld	9	239	2151
SIFT	695	32	22260

Table 1: Comparison of average per-image storage for the three techniques.

Fig. 5 and Table 1 present the precision-recall graph (average for querying on all images in the database) and the storage *efficiency* for each of the methods. We see from Table 1 that our method is the most efficient in the average amount of data it generates per image. For retrieval, we find the SIFT Keys outperform the other two methods. This result agrees with the study by Mikolajczyk and Schmid [17]. Our method outperforms the Blobworld technique by about 5% precision on average.

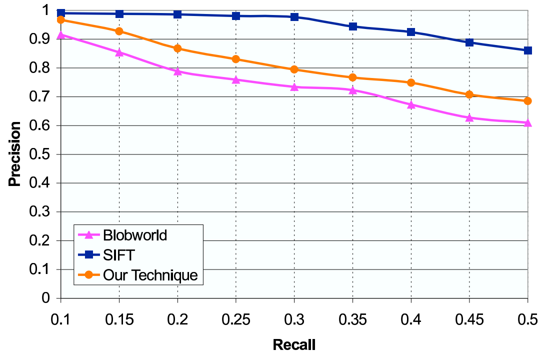

Figure 5: Comparison between our technique and other published techniques.

In § 3.5 we discussed the properties of our representation, and claimed that it is robust to affine transformations of the image. To test this, we halved the aspect ratio of the entire dataset and re-computed the coherent regions and the SIFT Keys. We performed a complete dataset query (same as above) and measured the precision-recall (Fig. 6) when querying with these distorted images. From the graph, we see that our method is very robust to the image distortion and it outperforms the SIFT method which drops substantially.

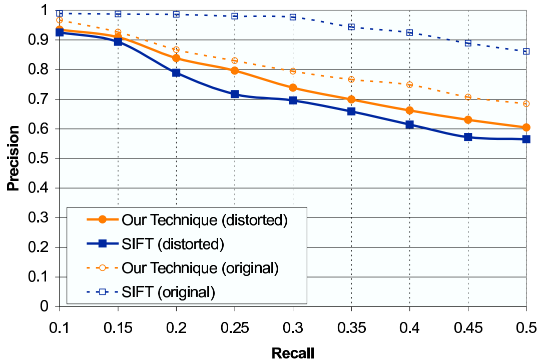

Figure 6: Graph showing precision-recall for our technique and the SIFT method when querying with distorted images from the database. The aspect ratios of the images were halved.

In Fig. 7, we show the effect of varying the number of projections used in the image description. For Proj. 1, we just use the grayscale image. For Proj. 2, we use the grayscale image and the variance projection with a neighborhood size of 32. For Proj. 3, we use the 3 opponent color axes, and for Proj. 4, we add the variance with neighborhood size 32. Proj. 5 is the same set of projections used in all the other experiments. We find that the addition of multiple projections greatly improves the retrieval accuracy. However, we note that there is not a large difference between Proj. 4 and Proj. 5. We claim this is because of the similarity in projections Proj. 4 and Proj. 5 which both just measure neighborhood variance but in different sized neighborhoods.

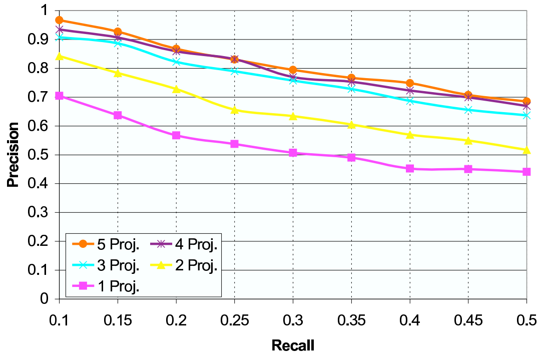

Figure 7: Graph showing precision-recall as the number of projections (feature spaces) is varied.

In Fig. 8, we show the effect of using kernel-weighted means for region description versus standard, uniformly weighted means. As expected, the kernel-weighted means greatly outperform the uniform means (by about 10% on average).

[†] This data reflects the available source code for Blobworld and SIFT. It should be noted that the SIFT keys store 128 1-byte elements while the other two methods use 4-byte (1-word) floating point elements. We have not experimented with quantizing the storage for our technique to further reduce the size.

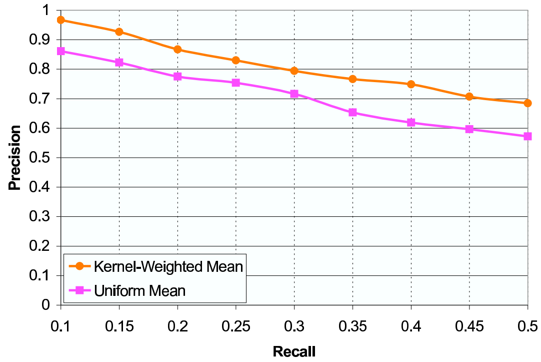

Figure 8: Graph showing precision-recall using kernel-weighted means in the projections versus uniform means.

6. Conclusion

We have presented a novel method for image representation using a kernel-based, sparse image segmentation and description method. The method is general in that it permits a variety of feature spaces which are represented as scalar image projections. Our main contribution is in the image description which is a middle-ground between local interest operator techniques from object recognition and global image segmentation techniques that cluster regions of homogeneous content.

We create a continuous scale-space of regions with coherent image content. The regions are robust under drastic viewpoint changes and varying photometric conditions. Our initial experiments indicate that the method is stable, reliable, and efficient in terms of both computation and storage. In particular, the use of spatial kernels admits efficient, optimization-based methods for segmentation and, ultimately, image registration.

There are several directions of future work we intend to pursue. The problem of registration using segmentation has been addressed by Schaffalitzky and Zisserman [20]. One advantage of kernel-based methods is that the registration problem can be posed as a continuous optimization defined directly on images. We intend to investigate this approach. A second, obvious extension is to enrich the set of feature spaces we consider. In particular, the development of suitably invariant projections for texture is a key step. Finally, a more extensive evaluation of our methods with a larger, more varied database is an essential further step.

Acknowledgments

This work is funded by the National Science Foundation under Grant No. 0112882. We thank the groups at UC Berkeley and David Lowe for providing copies of their source code/binaries.

References

[1] Chad Carson, Serge Belongie, Hayit Greenspan, and Jitendra Malik. Blobworld: Image Segmentation using Expectation-Maximization and Its Application to Image Querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, 2002.

[2] Robert Collins. Mean-Shift Blob Tracking Through Scale Space. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[3] Robert Collins and Yanxi Liu. On-Line Selection of Discriminative Tracking Features. In *International Conference on Computer Vision*, volume 1, pages 346–352, 2003.

[4] Dorin Comaniciu and Peter Meer. Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[5] Dorin Comaniciu, Visvanathan Ramesh, and Peter Meer. The Variable Bandwidth Mean Shift and Data-Driven Scale Selection. In *International Conference on Computer Vision*, volume 1, pages 438–445, 2001.

[6] Hayit Greenspan, Jacob Goldberger, and L. Ridel. A Continuous Probabilistic Framework for Image Matching. *Computer Vision and Image Understanding*, 84:384–406, 2001.

[7] Timor Kadir. *Scale, Saliency, and Scene Description*. PhD thesis, University of Oxford, 2002.

[8] Timor Kadir and Michael Brady. Saliency, Scale and Image Description. *International Journal of Computer Vision*, 43(2):83–105, 2001.

[9] Svetlana Lazebnik, C. Schmid, and Jean Ponce. Affine-Invariant Local Descriptors and Neighborhood Statistics for Texture Recognition. In *International Conference on Computer Vision*, pages 649–656, 2003.

[10] Tony Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers, 1994.

[11] Tony Lindeberg. Principles for Automatic Scale Selection. In *Academic Press, Boston, USA*, volume 2, pages 239–274. 1999.

[12] David Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[13] Jitendra Malik, Serge Belongie, Jianbo Shi, and Thomas Leung. Textons, Contours, and Regions: Cue Combination in Image Segmentation. In *International Conference on Computer Vision*, 1999.

[14] David Marr and E. Hildreth. Theory of Edge Detection. In *Royal Society of London B*, volume 290, pages 199–218, 1980.

[15] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In *British Machine Vision Conference*, 2002.

[16] J. Matas, S. Obdrzalek, and O. Chum. Local Affine Frames for Wide-Baseline Stereo. In *International Conference on Pattern Recognition*, 2002.

[17] K. Mikolajczyk and C. Schmid. A Performance Evaluation of Local Descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 257–264, 2003.

[18] Y. Ohta, Takeo Kanade, and T. Sakai. Color Information for Region Segmentation. *Computer Graphics and Image Processing*, 13(3):222–241, 1980.

[19] Kazunori Okada, Dorin Comaniciu, and Arun Krishnan. Scale Selection for Anisotropic Scale-Space: Application ot Volumetric Tumor Characterization. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 594–601, 2004.

[20] F. Schaffalitzky and Andrew Zisserman. Viewpoint Invariant Texture Matching and Wide Baseline Stereo. In *International Conference on Computer Vision*, volume 2, pages 636–643, 2001.

[21] C. Schmid and R. Mohr. Local Grayvalue Invariants for Image Retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, 1997.

[22] T. Tuytelaars and Luc Van Gool. Matching Widely Separated Views Based on Affine Invariant Regions. *International Journal of Computer Vision*, 59(1):61–85, 2004.