

# Adaptive Quasiconformal Kernel Metric for Image Retrieval

Douglas R. Heisterkamp  
Computer Science Dept.  
Oklahoma State University  
Stillwater, OK 74078  
doug@cs.okstate.edu

Jing Peng  
Electrical Engr. & Computer Sci.  
Tulane University  
New Orleans, LA 70118  
jp@eecs.tulane.edu

H. K. Dai  
Computer Science Dept.  
Oklahoma State University  
Stillwater, OK 74078  
dai@cs.okstate.edu

## Abstract

*This paper presents a new approach to ranking relevant images for retrieval. Distance in the feature space associated with a kernel is used to rank relevant images. An adaptive quasiconformal mapping based on relevance feedback is used to generate successive new kernels. The effect of the quasiconformal mapping is a change in the spatial resolution of the feature space. The spatial resolution around irrelevant samples is dilated, whereas the spatial resolution around relevant samples is contracted. This new space created by the quasiconformal kernel is used to measure the distance between the query and the images in the database. An interesting interpretation of the metric is found by looking at the Taylor series approximation to the original kernel. Then the squared distance in the feature space can be seen as a combination of a parzen window estimate of the squared Chi-squared distance and a weighted squared Euclidean distance. Experimental results using real-world data validate the efficacy of our method.*

## 1. Introduction

Relevance feedback is often used to allow a user to refine her query. A user labels the images from the previous query as relevant or irrelevant. From this feedback, the system attempts to learn the user's concept of relevance to the query. One approach to this learning is the creation of a similarity measure (or a distance). If the similarity measure reflects the user's concept of relevance then images with high similarity (or small distance) to the query have a high relevance to the user. This process iterates until the user is satisfied with the image retrieval or abandons the search.

This paper presents a new approach to learning the user's concept of relevance by creating an adaptive quasiconformal kernel distance metric. The layout of the paper is as follows. In Section 2, distance in the feature space created by a kernel is reviewed. In Section 3, a method for gen-

erating a quasiconformal kernel from relevance feedback is presented. In Section 4, the distance metric created by the adaptive quasiconformal kernel, AQK, is used for image retrieval and is experimentally compared with MARS[13] and PFRL[11].

### 1.1. Previous Work

A recent work [12] describes an image retrieval system (MARS) that makes use of retrieval techniques developed in the field of *information retrieval* (IR) for text-based information. In this system, images are represented by weight vectors in the term space, where weights capture the importance of components within a vector as well as importance across different vectors over the entire data set. The system then uses relevance feedback to update queries so as to place more weight on relevant terms and less weight on irrelevant ones. This query updating mechanism amounts to rotating the query vector toward relevant retrievals and, at the same time, away from irrelevant ones. One limitation of this system is that it is variant to translation and general linear transformation because of its use of the non-metric similarity function [5]. Another limitation with the technique is that in many situations mere query shifting is insufficient to achieve desired goals.

Peng et al. [11] present probabilistic feature relevance learning for content-based image retrieval that computes flexible retrieval metrics for producing neighborhoods that are elongated along less relevant feature dimensions and constricted along most influential ones. The technique has shown promise in a number of image database applications. The main difference between the retrieval metric proposed here and the one presented in [11] is that the former is capable of producing arbitrary, non-symmetric neighborhoods, while the latter cannot.

The MindReader system described in [8] uses a multi-level scoring technique to adaptively compute a distance matrix and new query locations. While it provides a theoretical basis for MARS, it fails to analyze its working con-

ditions. In addition, providing scores to retrieved images places burden on the user. In contrast, we exploit local likelihood information to adjust spatial resolution so that relevant images move toward each other and away from irrelevant ones.

In [12], MindReader is generalized to allow a distance matrix to be either diagonal or a full matrix, depending on the availability of data. While it often provides improvement in computation, it still requires full matrix computation even if input features are independent. Also, like MindReader, multi-level scoring potentially places strain on the user.

## 2. Kernel Distance

The kernel trick has been applied to numerous problems [16, 18, 4, 17, 9]. The kernel allows an algorithm to work in a feature space of higher dimension. If  $\phi(\mathbf{x})$  is a mapping of a point in the input space to the feature space, then the kernel calculates the dot product in the feature space of the image of two points from input space,  $\mathbf{K}(\mathbf{a}, \mathbf{b}) = \langle \phi(\mathbf{a}), \phi(\mathbf{b}) \rangle$ . Common kernels are Gaussian,  $\mathbf{K}_{\text{gaus}}(\mathbf{a}, \mathbf{b}) = e^{-\frac{\|\mathbf{a}-\mathbf{b}\|^2}{2\sigma^2}}$ , and polynomial,  $\mathbf{K}_{\text{poly}}(\mathbf{a}, \mathbf{b}) = (1 + \langle \mathbf{a}, \mathbf{b} \rangle)^d$ .

Distance in the feature space may be calculated by means of the kernel [19, 4]. With  $\mathbf{a}$  and  $\mathbf{b}$  in the input space then the squared feature space distance is

$$\begin{aligned} \text{dist}_{\mathbf{K}}(\mathbf{a}, \mathbf{b})^2 &= \|\phi(\mathbf{a}) - \phi(\mathbf{b})\|^2 \\ &= \mathbf{K}(\mathbf{a}, \mathbf{a}) - 2\mathbf{K}(\mathbf{a}, \mathbf{b}) + \mathbf{K}(\mathbf{b}, \mathbf{b}). \end{aligned} \quad (1)$$

Since the dimensionality of the feature space may be very high, the meaning of the distance is not directly apparent. Since the image of the input space forms a submanifold in the feature space with the same dimensionality as the input space, what is available is a Riemannian metric [1, 16, 4]. The Riemannian metric tensor induced by a kernel is (see, [4, p. 48])

$$g_{ij}(\mathbf{z}) = \left( \frac{1}{2} \frac{\partial^2 \mathbf{K}(\mathbf{x}, \mathbf{x})}{\partial x_i \partial x_j} - \frac{\partial^2 \mathbf{K}(\mathbf{x}, \mathbf{z})}{\partial x_i \partial x_j} \right)_{\mathbf{x}=\mathbf{z}}. \quad (2)$$

## 3. Quasiconformal Kernel

It is a straight forward process to create a new kernel from existing kernels [4]. Since it is our desire to create a new feature space in which the spatial resolution around relevant samples is contracted and the spatial resolution around irrelevant sample is expanded, we look to quasiconformal mappings [2]. Previously, [1] has modified a support vector machine with a quasiconformal mapping. An initial desire may be to use a conformal mapping. But in higher

dimensional space, conformal mapping are limited to similarity transformations and spherical inversions [3] and hence it may be difficult to find a conformal mapping with the properties we desire. Quasiconformal are more general than conformal mapping, containing conformal mappings as a special case. Hence, it is easier to find a quasiconformal mapping with the properties we desire. If  $c(\mathbf{x})$  is a positive real valued function of  $\mathbf{x}$  element of the input space, then a new kernel can be created by

$$\tilde{\mathbf{K}}(\mathbf{a}, \mathbf{b}) = c(\mathbf{a})c(\mathbf{b})\mathbf{K}(\mathbf{a}, \mathbf{b}). \quad (3)$$

Amari called this kernel a conformal transformed kernel. We are calling it a quasiconformal kernel. Note that if the restriction on  $c(\mathbf{x})$  being positive is removed, a still a valid kernel [4].

The question becomes which  $c(\mathbf{x})$  do we wish to use? We can change the Riemannian metric by the choice of  $c(\mathbf{x})$ . The metric  $g_{i,j}$  associated with kernel  $\mathbf{K}$  becomes the metric  $\tilde{g}_{i,j}$  associated with kernel  $\tilde{\mathbf{K}}$  by the relationship [1, Theorem 2,]

$$\tilde{g}_{i,j}(\mathbf{x}) = c_i(\mathbf{x})c_j(\mathbf{x}) + c(\mathbf{x})^2 g_{i,j}(\mathbf{x}) \quad (4)$$

where  $c_i(\mathbf{x}) = \frac{\partial c(\mathbf{x})}{\partial x_i}$ .

Amari expanded the spatial resolution in the margin of a support vector machine by using the following [1, Equation 26]

$$c(\mathbf{x}) = \sum_{i \in \mathcal{SV}} h_i e^{-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{2\tau^2}} \quad (5)$$

where  $\mathcal{SV}$  is the set of support vectors,  $h_i$  is a positive number representing the contribution of the  $i$ th support vector,  $\mathbf{x}_i$  is the  $i$ th support vector, and  $\tau$  is a free parameter. This can be viewed as a non-normalized parzen window density estimation of the set of support vectors. Thus the relative spatial resolution is dilated in area of high estimated density and contracted in areas of low estimated density. Since the support vectors are at the boundary of the margin, this creates an expansion of spatial resolution in the margin and a contraction elsewhere.

Our goal is to expand the spatial resolution around irrelevant samples and contract the spatial resolution around relevant samples. That is, distance to irrelevant samples is increased and distance to relevant samples are decreased. We want  $c(\mathbf{x})$  to adapt to the relevance feedback. Looking at Amari's approach as a density estimation of the set of support vectors, we can estimate the density of the set of irrelevant samples,  $\mathcal{I}$ , and the set of relevant samples,  $\mathcal{R}$ .

The parzen window estimate [5] using the relevance feedback is

$$\hat{P}(\mathbf{x}|I) = \frac{1}{|\mathcal{I}|} \sum_{\mathbf{x}_i \in \mathcal{I}} e^{-\frac{\|\mathbf{x}-\mathbf{x}_i\|^2}{2\tau^2}}, \quad (6)$$

and

$$\hat{P}(\mathbf{x}|R) = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{x}_i \in \mathcal{R}} e^{-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{2\tau^2}}. \quad (7)$$

We construct our quasiconformal mapping from the parzen window estimates by defining  $c(\mathbf{x})$  as

$$c(\mathbf{x}) = \frac{\hat{P}(\mathbf{x}|I)}{\hat{P}(\mathbf{x}|R)}. \quad (8)$$

Other possible choices include  $c(\mathbf{x}) = \tan(\frac{(1+z)\pi}{4})$  and  $c(\mathbf{x}) = \frac{(1+z)}{(1-z)}$  where  $z = \hat{P}(\mathbf{x}|I) - \hat{P}(\mathbf{x}|R)$ . The meaning and numerical stability of other possible forms for  $c(\mathbf{x})$  will be investigated in a future work.

The adaptive quasiconformal kernel metric is

$$\begin{aligned} \text{dist}_{\text{aqk}}(\mathbf{x}, \mathbf{q})^2 &= \tilde{\mathbf{K}}(\mathbf{x}, \mathbf{x}) - 2\tilde{\mathbf{K}}(\mathbf{x}, \mathbf{q}) + \tilde{\mathbf{K}}(\mathbf{q}, \mathbf{q}) \\ &= c(\mathbf{x})^2 \mathbf{K}(\mathbf{x}, \mathbf{x}) - 2c(\mathbf{x})c(\mathbf{q})\mathbf{K}(\mathbf{x}, \mathbf{q}) + c(\mathbf{q})^2 \mathbf{K}(\mathbf{q}, \mathbf{q}). \end{aligned}$$

If the original kernel,  $\mathbf{K}$  is a Gaussian kernel, then  $\mathbf{K}(\mathbf{a}, \mathbf{a}) = 1$  and the distance metric becomes

$$\text{dist}_{\text{aqk}}(\mathbf{x}, \mathbf{q})^2 = c(\mathbf{x})^2 - 2c(\mathbf{x})c(\mathbf{q})\mathbf{K}(\mathbf{x}, \mathbf{q}) + c(\mathbf{q})^2. \quad (9)$$

An interesting interpretation of this distance can be obtained if we look at the distance using a first order Taylor expansion of the Gaussian kernel at the query point,  $\mathbf{q}$ ,

$$\mathbf{K}_{\text{gaus}}(\mathbf{x}, \mathbf{q}) \cong 1 - \frac{1}{2\sigma^2} \|\mathbf{x} - \mathbf{q}\|^2. \quad (10)$$

Substituting the Taylors expansion into (9) yields

$$\text{dist}_{\text{aqc}}(\mathbf{x}, \mathbf{q})^2 = (c(\mathbf{x}) - c(\mathbf{q}))^2 + \frac{c(\mathbf{x})c(\mathbf{q})}{\sigma^2} \|\mathbf{x} - \mathbf{q}\|^2 \quad (11)$$

The  $(c(\mathbf{x}) - c(\mathbf{q}))^2$  term can be viewed as the weighted squared Chi-squared distance between points  $\mathbf{x}$  and  $\mathbf{q}$ . The Chi-squared distance [6], is

$$\text{dist}_{\text{Chi}^2}(\mathbf{x}, \mathbf{q})^2 = \sum_{j=1}^J \frac{(P(j|\mathbf{x}) - P(j|\mathbf{q}))^2}{P(j|\mathbf{q})}. \quad (12)$$

The term  $(c(\mathbf{x}) - c(\mathbf{q}))^2$  is

$$\left( \frac{P(\mathbf{x}|I)}{P(\mathbf{x}|R)} - \frac{P(\mathbf{q}|I)}{P(\mathbf{q}|R)} \right)^2 = \frac{P(R)^2}{P(I)^2} \left( \frac{P(I|\mathbf{x})}{P(R|\mathbf{x})} - \frac{P(I|\mathbf{q})}{P(R|\mathbf{q})} \right)^2$$

and thus is related to a weighted Chi-squared distance. The second term is a weighted Euclidean distance in the input space.



Figure 1. Example images from texture data

## 4. Experimental Results

In the following, we compare the retrieval method based on the adaptive quasiconformal kernel metric, AQK, with MARS [13] and PFRL[11]. Performance is measured by *average precision* [15, 14] and by the *precision-recall* graph.

**Average Precision:** The non-interpolated average precision is defined as

$$\frac{1}{|\mathcal{Rel}|} \sum_{d \in \mathcal{Rel}} \frac{|\{d' | d' \in \mathcal{Rel}, \text{Rank}(d') \leq \text{Rank}(d)\}|}{\text{Rank}(d)} \quad (13)$$

where  $\mathcal{Rel}$  is the set of relevant items in the data set, and Rank is an ordering of the data set based on similarity with the query.

**Precision-Recall graph:** For an item,  $d$ , in the data set, the recall at  $d$  is the ratio of the number of relevant elements returned when  $d$  is reached to the total number of relevant elements in the data set.

$$\text{Recall}(d) = \frac{|\{d' | d' \in \mathcal{Rel}, \text{Rank}(d') \leq \text{Rank}(d)\}|}{|\mathcal{Rel}|} \quad (14)$$

For an item,  $d$ , in the data set, the precision at  $d$  is the ratio of the number of relevant elements returned when  $d$  is reached to the total number of elements returned when  $d$  is reached.

$$\text{Precision}(d) = \frac{|\{d' | d' \in \mathcal{Rel}, \text{Rank}(d') \leq \text{Rank}(d)\}|}{\text{Rank}(d)} \quad (15)$$

The precision-recall graph is a line graph of the points  $(\text{Recall}(d), \text{Precision}(d))$  for every element  $d$  of the data set. Both high recall and high precision is desired, though not often obtainable.

The following four data set were used for evaluation:

**Texture** — the texture data are obtained from MIT Media Lab at: [whitechapel.media.mit.edu/pub/VisTex](http://whitechapel.media.mit.edu/pub/VisTex). There are a total of 640 images of  $128 \times 128$  in the database with 15 classes. The images in this database are represented by 8 Gabor filters (2 scales and 4 orientations). Examples of the textures are presented in Figure 1.

**Vowel** — the vowel data set has  $q = 10$  measurements and 11 classes. There are a total of  $N = 528$  samples in this database. This data set is also taken from [10].

**Segmentation** The segmentation data set, taken from the UCI repository [10], consists of images that were drawn randomly from a database of 7 outdoor images. The images were hand-segmented by the creators of the database to classify each pixel. Each image is a region. There are 7 classes, each of which has 330 instances. Thus, there are a total of 2310 images in the database. These images are represented by 19 real valued attributes.

**Letter** — the letter data set consists of  $n = 16$  numerical attributes and 26 classes of the two capital letters, O and Q, in the English alphabet. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli.

To determine the free parameters in AQK, ( $\sigma$  and  $\tau$ ), a ten-fold cross-validation was ran for for the Texture, Vowel, and Segmentation data sets. Each data set was divided into ten partitions. Each partition in turn was left out and the other nine were used to determine values for  $\sigma$  and  $\tau$ . The left out partition was then used to test the algorithm. The values reported are the average of the ten tests.

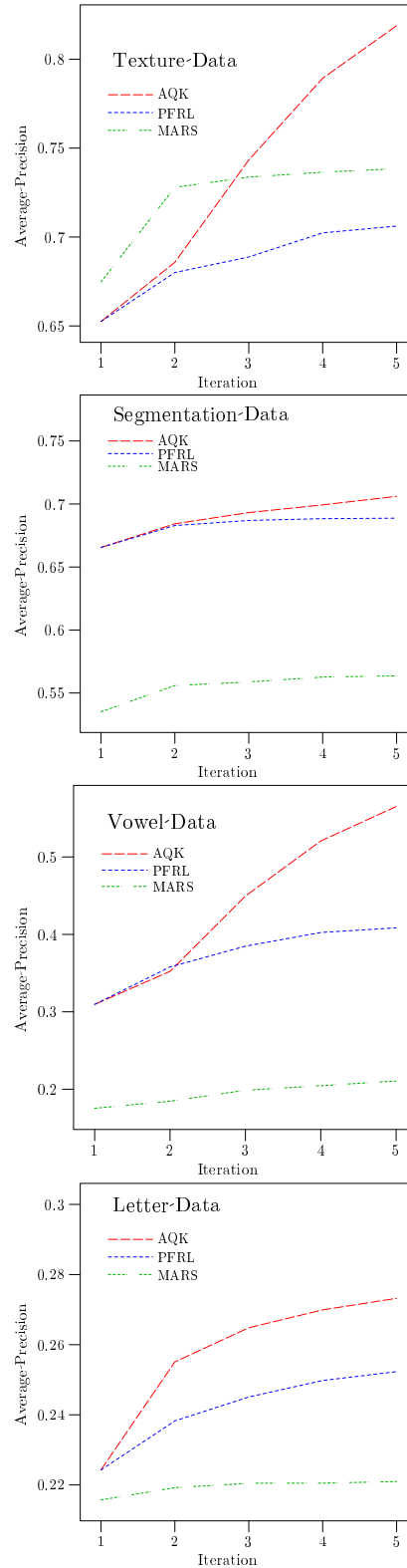
Ten-fold cross validation was not used on the Letter data set due to time constraints. A test set of 100 randomly selected elements from each class was created (2600 items). From remaining elements, ten randomly selected elements from each class( 260 items) were used to determining the parameters.

The average precision versus iteration of relevance feedback is presented in Figure 2. The AQK algorithm performs well on the data sets. The most striking aspect is the continued improvement in AQK’s average precision in the Texture and Vowel data sets where as MARS’s and PFRL’s improvement flattens.

The precision-recall graphs of the three algorithms for the Texture data set is presented in Figure 3, for Segmentation data set in Figure 4, for Vowel data set in Figure 5, and for Letter data set in Figure 6. The AQK algorithm has the desirable effect of pulling the precision-recall curve towards the upper right with successive iterations of relevance feedback. Again, this is most noticeable in the Texture and Vowel data sets.

## 5. Summary

A quasiconformal kernel constructed by relevance feedback was used to create a metric for use in ranking relevant images of a query. The metric is the Euclidean distance in the feature space associated with the quasiconformal kernel. The quasiconformal kernel is constructed in such a way that around relevant labelled images the spatial resolution is contracted and that around irrelevant labelled images the



**Figure 2. Average Precision versus iteration of relevance feedback**

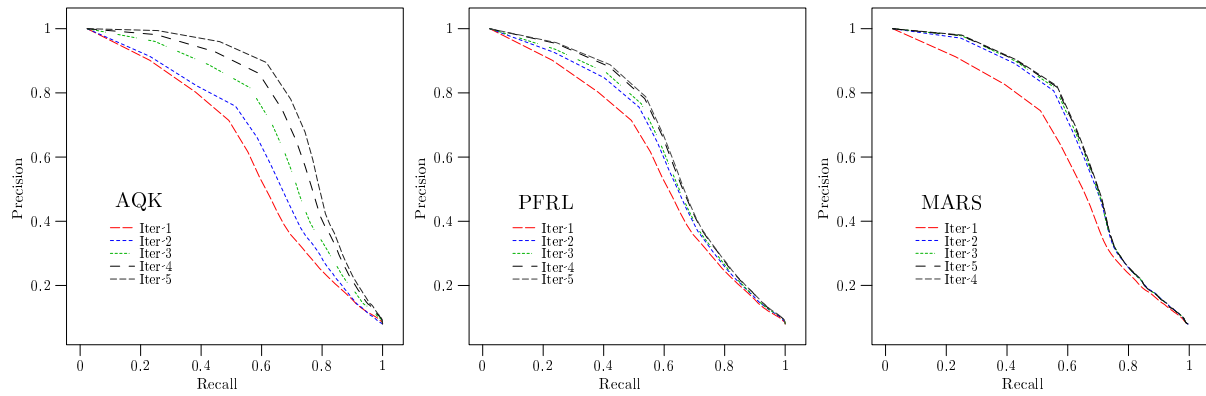


Figure 3. Precision vs. Recall — Texture Data

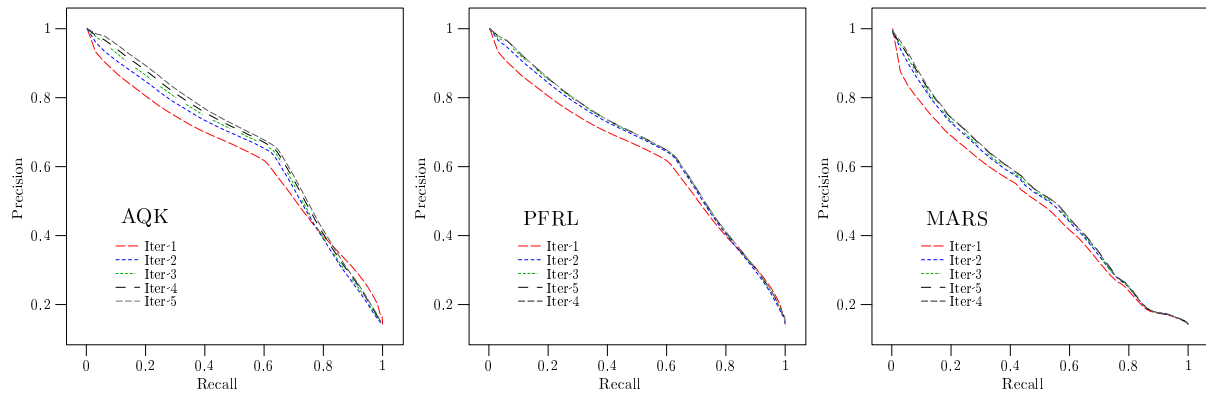


Figure 4. Precision vs. Recall — Segmentation Data

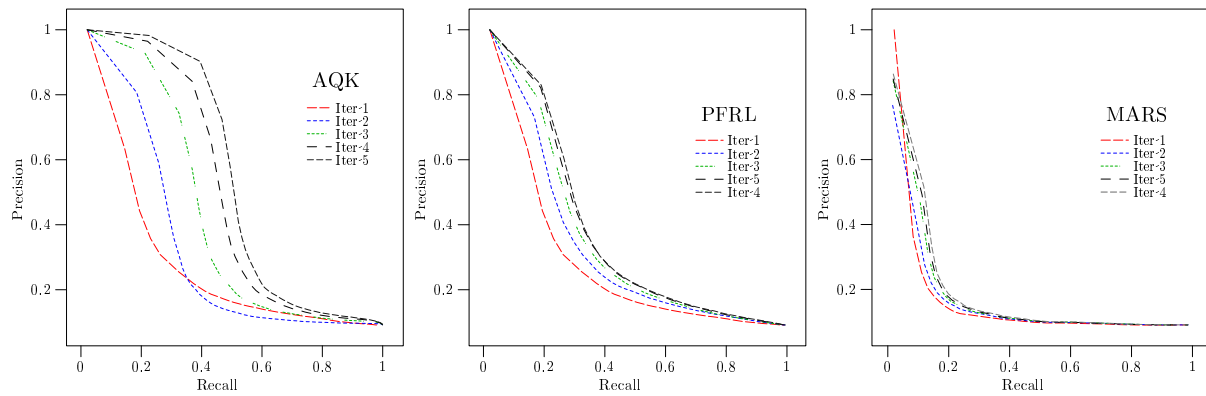


Figure 5. Precision vs. Recall — Vowel Data

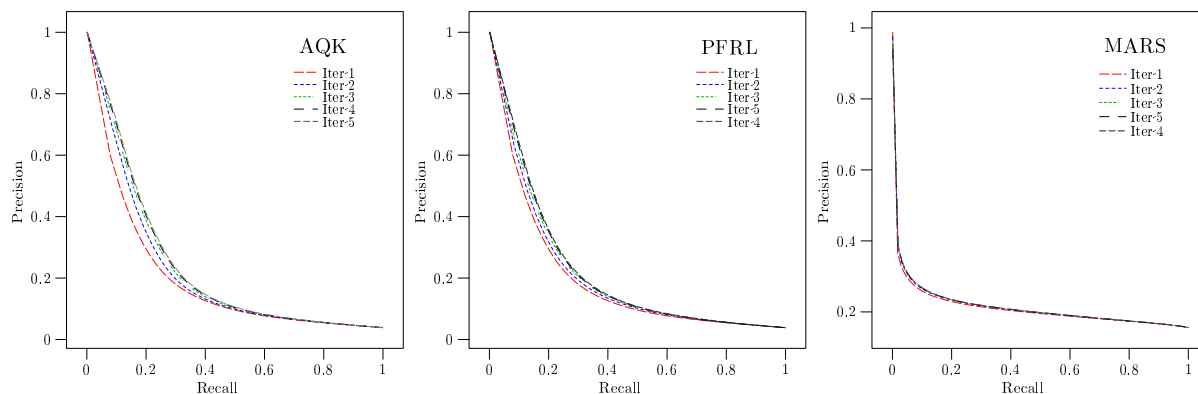


Figure 6. Precision vs. Recall — Letter Data

spatial resolution is diluted. The distance from the query to unlabeled images is measured in this new space. The metric generated by AQK has an interpretation in the input space as a combination of a weighted Chi-squared distance and a weighted Euclidean distance. In application, the new AQK approach compares very well with the existing MARS and PFRL approaches.

Future work includes an investigation of the meaning and numerical stability of other quasiconformal mapping. In addition, for AQK to be practical, an efficient database indexing using the AQK metric needs to be developed.

## References

- [1] S. Amari and S. Wu. Improving support vector machine classifiers by modifying kernel functions. *Neural Networks*, 12(6):783–789, 1999.
- [2] G. D. Anderson, M. K. Vananamurthy, and M. K. Vuorinen. *Conformal Invariants, Inequalities, and Quasiconformal Maps*. Canadian Mathematical Society Series of Monographs and Advanced Texts. John Wiley & Sons, Inc., New York, 1997.
- [3] D. E. Blair. *Inversion Theory and Conformal Mapping*. American Mathematical Society, 2000.
- [4] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, Cambridge, UK, 2000.
- [5] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, Inc., New York, second edition, 2001.
- [6] T. Hastie and R. Tibshirani. Discriminant adaptive nearest neighbor classification and regression. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 409–415. The MIT Press, 1996.
- [7] T. Huang and Y. Rui. Image retrieval: Past, present, and future. In *Proc. of Int. Symposium on Multimedia Information Processing*, Dec 1997.
- [8] Y. Ishikawa, R. Subramanya, and C. Faloutsos. MindReader: Querying databases through multiple examples. In *Proceedings of 24th International Conference on Very Large Data Bases*, pages 218–227, 24–27 1998.
- [9] K. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–201, March 2001.
- [10] P. Murphy and D. Aha. Uci repository of machine learning databases. [www.cs.uci.edu/~mlearn/MLRepository.html](http://www.cs.uci.edu/~mlearn/MLRepository.html).
- [11] J. Peng, B. Bhanu, and S. Qing. Probabilistic feature relevance learning for content-based image retrieval. *Computer Vision and Image Understanding*, 75(1/2):150–164, 1999.
- [12] Y. Rui and T. Huang. Optimizing learning in image retrieval. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina*, pages 236–243, 2000.
- [13] Y. Rui, T. Huang, and S. Mehrotra. Content-based image retrieval with relevance feedback in mars. In *Proceedings of IEEE International Conference on Image Processing, Santa Barbara, California*, pages 815–818, October 1997.
- [14] R. E. Schapire and Y. Singer. Boostexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000.
- [15] R. E. Schapire, Y. Singer, and A. Singhal. Boosting and Rocchio applied to text filtering. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval*, pages 215–223, Melbourne, AU, 1998. ACM Press, New York, US.
- [16] B. Scholkopf, C. J. C. Burges, and A. J. Smola, editors. *Advances in kernel methods : support vector learning*. MIT Press, Cambridge, MA, 1999.
- [17] B. Scholkopf and etal. Input space versus feature space in kernel-based methods. *IEEE Transactions on Neural Networks*, 10(5):1000–1017, September 1999.
- [18] A. J. Smola, P. L. Bartlett, B. Scholkopf, and D. Schuurmans, editors. *Advances in Large Margin Classifiers*. MIT Press, Cambridge, MA, 2000.
- [19] V. N. Vapnik. *Statistical learning theory*. Adaptive and learning systems for signal processing, communications, and control. Wiley, New York, 1998.