



# Kernel Entropy Discriminant Analysis for Dimension Reduction

Aditya Mehta<sup>(✉)</sup>  and C. Chandra Sekhar 

Department of Computer Science and Engineering, Indian Institute of Technology  
Madras, Chennai, India

aditya.mhta@gmail.com, chandra@cse.iitm.ac.in

**Abstract.** The unsupervised techniques for dimension reduction, such as principal component analysis (PCA), kernel PCA and kernel entropy component analysis, do not take the information about class labels into consideration. The reduced dimension representation obtained using the unsupervised techniques may not capture the discrimination information. The supervised techniques, such as multiple discriminant analysis and generalized discriminant analysis, can capture discriminatory information. However the reduced dimension is limited by number of classes. We propose a supervised technique, kernel entropy discriminant analysis (kernel EDA), that uses Euclidean divergence as criterion function. Parzen window method for density estimation is used to find an estimate of Euclidean divergence. Euclidean divergence estimate is expressed in terms of eigenvectors and eigenvalues of the kernel gram matrix. The eigenvalues and eigenvectors that contribute significantly to the Euclidean divergence estimate are used for determining the directions for projection. Effectiveness of the kernel EDA method is demonstrated through the improved classification accuracy for benchmark datasets.

**Keywords:** Euclidean divergence · Parzen windowing

## 1 Introduction

The goal of data transformation techniques is to transform a set of large number of features into a compact set of informative features. The data transformation techniques are unsupervised or supervised. In the unsupervised techniques, the aim is to preserve some significant characteristics of the data in the transformed space. The most commonly used unsupervised technique is the principal component analysis (PCA) [4]. It projects a given  $d$ -dimensional feature vector onto the eigenvectors of data covariance matrix corresponding to  $l$  most significant eigenvalues of the matrix. Kernel PCA [11] performs PCA in the kernel feature space of a Mercer kernel. The unsupervised techniques do not make use of class labels because of which the transformed representation may not be discriminative. In supervised data transformation techniques, the class label information is also used. The commonly used supervised techniques are Fisher discriminant analysis (FDA) [3], multiple discriminant analysis (MDA) [2] and their many

variants. The FDA finds a direction for projection along which the separability of projections of data belonging to two classes is maximum. The MDA is a multi-class extension of FDA. Kernel FDA [7] performs the FDA in the kernel feature space. Generalized discriminant analysis (GDA) [1] is extension of kernel FDA for multiple classes. A major limitation of these supervised techniques is that the dimension of transformed data is limited by the number of classes.

In the kernel entropy component analysis (kernel ECA) [5,6] technique, the eigenvectors of kernel gram matrix used for projection are determined based on their contribution to the Renyi quadratic entropy of the input data. The kernel ECA is an unsupervised technique.

In this paper, we develop a new discriminative transformation method that can be considered as an extension of kernel ECA modified for supervised dimension reduction. It chooses the directions for projection that maximally preserves the Euclidean divergence [10] between the probability density function of two classes. An estimator of Euclidean divergence is expressed in terms of eigenvectors and eigenvalues of kernel gram matrix of Gaussian kernel used in Parzen window [9] method for density estimation. The directions for projection are obtained using eigenvalues and eigenvectors that contribute significantly to the divergence estimate.

The paper is organized as follows. In Sect. 2, we present the kernel ECA method. We discuss the proposed kernel EDA method in Sect. 3. Experimental studies and results are presented in Sect. 4.

## 2 Kernel Entropy Component Analysis

Kernel entropy component analysis (Kernel ECA) focuses on entropy components instead of principal components that represent variance in kernel PCA. The Renyi's quadratic entropy of a distribution  $p(\mathbf{x})$  is given by

$$H(p(\mathbf{x})) = -\log \int p^2(\mathbf{x}) d\mathbf{x} \quad (1)$$

The information potential of a distribution  $p(\mathbf{x})$  is defined as  $V(p(\mathbf{x})) = \int p^2(\mathbf{x}) d\mathbf{x}$ . The information potential can also be expressed as  $V(p) = \mathcal{E}_p(p)$ , where  $\mathcal{E}_p(\cdot)$  denotes expectation w.r.t.  $p(\mathbf{x})$ . Consider the data set  $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ . Let  $k_\sigma(\mathbf{x}_m, \cdot)$  be the Gaussian kernel with  $\sigma$  as width of kernel used in the Parzen window method for estimation of density at  $\mathbf{x}_m$ . It may be noted that the Gaussian kernel is a Mercer kernel and, therefore, it is a positive semi-definite kernel. Then the estimate of density is given by

$$\hat{p}(\mathbf{x}_m) = \frac{1}{N} \sum_{\mathbf{x}_n \in D} k_\sigma(\mathbf{x}_m, \mathbf{x}_n) \quad (2)$$

Then the estimate of  $V(p(\mathbf{x}))$  denoted by  $\hat{V}(p)$  is given by

$$\hat{V}(p) = \frac{1}{N} \sum_{\mathbf{x}_m \in D} \hat{p}(\mathbf{x}_m) = \frac{1}{N} \sum_{\mathbf{x}_m \in D} \frac{1}{N} \sum_{\mathbf{x}_n \in D} k_\sigma(\mathbf{x}_m, \mathbf{x}_n) = \frac{1}{N^2} \mathbf{1}^T \mathbf{K} \mathbf{1} \quad (3)$$

where  $\bar{\mathbf{1}}$  is an  $(N \times 1)$  vector consisting of all 1's and  $\mathbf{K}$  is the kernel gram matrix of the kernel  $k_\sigma(\cdot, \cdot)$  on the dataset  $D$ . Using the eigen decomposition of kernel gram matrix  $\mathbf{K}$ , Eq. (3) can be rewritten as

$$\hat{V}(p) = \frac{1}{N^2} \sum_{i=1}^N (\sqrt{\lambda_i} \mathbf{e}_i^T \bar{\mathbf{1}})^2 \quad (4)$$

where  $\lambda_i$  and  $\mathbf{e}_i$  are the eigenvalues and eigenvectors of  $\mathbf{K}$  respectively. The eigenvectors of  $\mathbf{K}$  used for projection are identified based on their extent of contribution to the information potential estimate. It is noted that the kernel ECA method does not make use of the information about the class labels of examples in  $D$ .

### 3 Kernel Entropy Discriminant Analysis

Let  $D$  be the data set that consists of data of two classes,  $D_1 = \{\mathbf{x}_{11}, \mathbf{x}_{12}, \dots, \mathbf{x}_{1N_1}\}$  and  $D_2 = \{\mathbf{x}_{21}, \mathbf{x}_{22}, \dots, \mathbf{x}_{2N_2}\}$  which we assume are generated from probability density functions (pdf)  $p_1(\mathbf{x})$  and  $p_2(\mathbf{x})$  of classes respectively. The Euclidean divergence between the pdfs of these two classes is given as

$$ED(p_1, p_2) = \int p_1^2(\mathbf{x}) d\mathbf{x} - 2 \int p_1(\mathbf{x}) p_2(\mathbf{x}) d\mathbf{x} + \int p_2^2(\mathbf{x}) d\mathbf{x} \quad (5)$$

Using the Parzen window technique for pdf estimation the estimate of  $ED(p_1, p_2)$  denoted by  $\hat{ED}(p_1, p_2)$  is given by

$$\hat{ED}(p_1, p_2) = \frac{1}{N_1^2} \sum_{\mathbf{x}_m, \mathbf{x}_n \in D_1} k_\sigma(\mathbf{x}_m, \mathbf{x}_n) - \frac{2}{N_1 N_2} \sum_{\substack{\mathbf{x}_m \in D_1, \\ \mathbf{x}_n \in D_2}} k_\sigma(\mathbf{x}_m, \mathbf{x}_n) + \frac{1}{N_2^2} \sum_{\mathbf{x}_m, \mathbf{x}_n \in D_2} k_\sigma(\mathbf{x}_m, \mathbf{x}_n) \quad (6)$$

Let  $\mathbf{z}_1 = [z_{11}, z_{12}, \dots, z_{1N}]^T$  and  $\mathbf{z}_2 = [z_{21}, z_{22}, \dots, z_{2N}]^T$  be vectors with  $z_{ij} = 1$  if  $\mathbf{x}_j \in D_i$  and  $z_{ij} = 0$  otherwise. Thus Eq. (6) can be written as

$$\hat{ED}(p_1, p_2) = \frac{1}{N_1^2} \mathbf{z}_1^T \mathbf{K} \mathbf{z}_1 - \frac{2}{N_1 N_2} \mathbf{z}_1^T \mathbf{K} \mathbf{z}_2 + \frac{1}{N_2^2} \mathbf{z}_2^T \mathbf{K} \mathbf{z}_2 \quad (7)$$

where  $\mathbf{K}$  is the kernel gram matrix of the dataset  $D$ . Using the eigen decomposition of  $\mathbf{K}$ , Eq. (7) can be rewritten as

$$\hat{ED}(p_1, p_2) = \frac{1}{N_1^2} \sum_{i=1}^N (\sqrt{\lambda_i} \mathbf{e}_i^T \mathbf{z}_1)^2 - \frac{2}{N_1 N_2} \sum_{i=1}^N (\mathbf{e}_i^T \mathbf{z}_1) \lambda_i (\mathbf{e}_i^T \mathbf{z}_2) + \frac{1}{N_2^2} \sum_{i=1}^N (\sqrt{\lambda_i} \mathbf{e}_i^T \mathbf{z}_2)^2 \quad (8)$$

where  $\lambda_i$  are the eigenvalues and  $\mathbf{e}_i$  are the eigenvectors of  $\mathbf{K}$ . We can write Eq. (8) as

$$\hat{ED}(p_1, p_2) = \sum_{i=1}^N \lambda_i \left( \frac{\mathbf{e}_i^T \mathbf{z}_1}{N_1} - \frac{\mathbf{e}_i^T \mathbf{z}_2}{N_2} \right)^2 \quad (9)$$

Let  $\psi_i = \lambda_i \left( \frac{\mathbf{e}_i^T \mathbf{z}_1}{N_1} - \frac{\mathbf{e}_i^T \mathbf{z}_2}{N_2} \right)^2$ . Then  $\hat{ED}(p_1, p_2) = \sum_{i=1}^N \psi_i$

The term  $\psi_i$  is a measure of extent of contribution of  $\lambda_i$  and  $\mathbf{e}_i$  to the Euclidean divergence. Certain eigenvalues and eigenvectors for which  $\psi_i$  is large contribute more to the Euclidean divergence than the others. By considering only those eigenvalue and eigenvector pairs that contribute significantly to the divergence, we identify the directions for projection that can capture discriminatory features for the data of two given classes.

As the Gaussian kernel is also a Mercer kernel, it is an inner product kernel. Therefore,  $k_\sigma(\mathbf{x}_m, \mathbf{x}_n) = \langle \phi(\mathbf{x}_m), \phi(\mathbf{x}_n) \rangle$ , where  $\phi(\mathbf{x})$  is the kernel space representation of a data point  $\mathbf{x}$ . The mean vectors for two classes in  $\phi(x)$ -space are given by  $\mathbf{m}_1^\phi = \frac{1}{N_1} \sum_{\mathbf{x}_m \in D_1} \phi(\mathbf{x}_m)$  and  $\mathbf{m}_2^\phi = \frac{1}{N_2} \sum_{\mathbf{x}_m \in D_2} \phi(\mathbf{x}_m)$ . Then each term in Eq. (6) can be expressed as follows:

$$\frac{1}{N_1^2} \sum_{\mathbf{x}_m, \mathbf{x}_n \in D_1} k_\sigma(\mathbf{x}_m, \mathbf{x}_n) = \frac{1}{N_1^2} \sum_{\mathbf{x}_m, \mathbf{x}_n \in D_1} \langle \phi(\mathbf{x}_m), \phi(\mathbf{x}_n) \rangle = \langle \mathbf{m}_1^\phi, \mathbf{m}_1^\phi \rangle \quad (10)$$

$$\frac{1}{N_2^2} \sum_{\mathbf{x}_m, \mathbf{x}_n \in D_2} k_\sigma(\mathbf{x}_m, \mathbf{x}_n) = \frac{1}{N_2^2} \sum_{\mathbf{x}_m, \mathbf{x}_n \in D_2} \langle \phi(\mathbf{x}_m), \phi(\mathbf{x}_n) \rangle = \langle \mathbf{m}_2^\phi, \mathbf{m}_2^\phi \rangle \quad (11)$$

$$\frac{1}{N_1 N_2} \sum_{\substack{\mathbf{x}_m \in D_1, \\ \mathbf{x}_n \in D_2}} k_\sigma(\mathbf{x}_m, \mathbf{x}_n) = \frac{1}{N_1 N_2} \sum_{\substack{\mathbf{x}_m \in D_1, \\ \mathbf{x}_n \in D_2}} \langle \phi(\mathbf{x}_m), \phi(\mathbf{x}_n) \rangle = \langle \mathbf{m}_1^\phi, \mathbf{m}_2^\phi \rangle \quad (12)$$

Therefore, the estimate of Euclidean divergence  $\hat{ED}(p_1, p_2)$  can be expressed as

$$\hat{ED}(p_1, p_2) = \langle \mathbf{m}_1^\phi, \mathbf{m}_1^\phi \rangle - 2\langle \mathbf{m}_1^\phi, \mathbf{m}_2^\phi \rangle + \langle \mathbf{m}_2^\phi, \mathbf{m}_2^\phi \rangle = \|\mathbf{m}_1^\phi - \mathbf{m}_2^\phi\|^2 \quad (13)$$

Thus the Euclidean divergence in  $\mathbf{x}$  space corresponds to squared Euclidean distance between the means of the data of two classes in the kernel feature space. Let  $\boldsymbol{\nu}_i$  be the direction for projections in the  $\phi(\mathbf{x})$ -space. As in kernel PCA, the vector  $\boldsymbol{\nu}_i$  is expressed as follows:

$$\boldsymbol{\nu}_i = \frac{1}{\sqrt{\lambda_i}} \sum_{n=1}^N e_{in} \phi(\mathbf{x}_n) \quad (14)$$

where  $e_{in}$  is the  $n^{\text{th}}$  element of  $\mathbf{e}_i$ . The projection of a given data point  $\phi(\mathbf{x})$  in the kernel feature space is given by

$$a_i = \boldsymbol{\nu}_i^T \phi(\mathbf{x}) = \frac{1}{\sqrt{\lambda_i}} \sum_{n=1}^N e_{in} \phi(\mathbf{x}_n)^T \phi(\mathbf{x}) = \frac{1}{\sqrt{\lambda_i}} \sum_{n=1}^N e_{in} k(\mathbf{x}_n, \mathbf{x}), \quad (15)$$

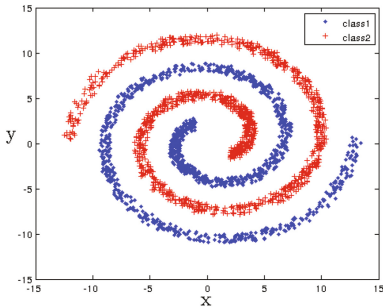
For a given data point  $\mathbf{x}$ , the  $l$ -dimensional transformed representation using the kernel EDA method is obtained by computing  $a_i$ ,  $i = 1, 2, \dots, l$  where  $l$  is the number of directions for projection chosen.

## 4 Experiments

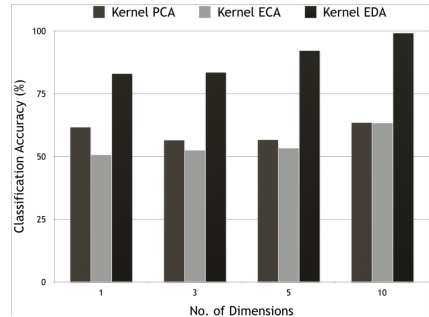
First, we analyze our proposed algorithm on a synthetic dataset. Then we evaluate the performance on two-class IDA benchmark datasets and multi-class datasets. We compare the performance of the proposed kernel EDA method with PCA, kernel PCA, kernel ECA, kernel FDA and GDA. We have used the Gaussian kernel for the data transformation techniques. The kernel width  $\sigma$  is chosen empirically for each dataset and in order to make a fair comparison, the same kernel width is used for all the techniques. Linear support vector machine is used for obtaining the classification accuracy on the transformed representation of data. The choice of using a linear classifier helps us identifying how effective is the transformed representation in performing the classification task. The data is split into 75%, 10% and 15% for training, validation and testing respectively.

### 4.1 Studies on Synthetic Dataset

The synthetic dataset contains 3424 data points randomly distributed on two spirals, shown in Fig. 1(a). The classification accuracies on the transformed data using linear SVM are plotted in Fig. 1(b). The accuracies are shown for kernel PCA, kernel ECA and kernel EDA methods and for different values of  $l$ . It is seen that kernel EDA performs better than the other two methods.



(a) Synthetic spiral dataset



(b) Classification accuracy

**Fig. 1.** (a) Synthetic spiral dataset (b) Classification accuracies (in %) for different methods for data transformation and for different values of transformed dimensions.

**Table 1.** Details of the 2-class IDA benchmark datasets used.

Name	No. of Examples	Dimension
Breast Cancer	263	9
Diabetes	768	8
German	1000	20
Image	2086	18
Splice	2991	60
Waveform	5000	21

## 4.2 Studies on 2-Class Real World Datasets

Details of two-class IDA benchmark datasets [6] used in our study are listed in Table 1. The classification accuracies on transformed representation obtained using different methods are given in Table 2. The results show an improvement in performance for the proposed technique over the existing techniques on almost all of the datasets.

**Table 2.** Classification accuracies (in %) obtained on the 2-class IDA benchmark datasets for different methods of dimensionality reduction and for different values of reduced dimension.

Dataset	Reduced dimension ( $l$ )	PCA	Kernel PCA	Kernel ECA	Kernel FDA	Kernel EDA
Breast Cancer	1	<b>69.7</b>	68.2	68.2	68.1	68.2
	3	68.2	<b>69.7</b>	68.2	-	68.2
	5	69.7	69.7	69.7	-	<b>72.7</b>
	7	69.7	69.7	69.7	-	<b>72.7</b>
Diabetes	1	<b>69.8</b>	65.1	59.8	61.4	59.4
	3	<b>67.2</b>	66.1	66.1	-	<b>67.2</b>
	5	68.2	67.7	67.7	-	<b>71.4</b>
	7	72.9	75.5	75.5	-	<b>77.1</b>
German	1	<b>69.2</b>	<b>69.2</b>	<b>69.2</b>	<b>69.2</b>	<b>69.2</b>
	3	<b>69.2</b>	<b>69.2</b>	<b>69.2</b>	-	<b>69.2</b>
	5	71.6	73.2	74.0	-	<b>74.4</b>
	10	<b>76.0</b>	75.2	<b>76.0</b>	-	<b>76.0</b>
	15	75.2	74	74.4	-	<b>75.6</b>
Image	1	57.8	58.3	58.3	58.3	<b>69.9</b>
	3	71.0	70.8	69.7	-	<b>75.8</b>
	5	69.3	71.0	67.2	-	<b>80.2</b>
	10	72.2	82.7	82.7	-	<b>85.8</b>
	15	83.4	83.8	82.9	-	<b>87.7</b>
Splice	1	69.8	69.7	60.0	86.3	<b>75.0</b>
	3	67.2	80.6	69.1	-	<b>77.5</b>
	5	68.2	<b>82.1</b>	80.3	-	80.3
	10	66.1	81.3	81.1	-	<b>83.7</b>
	15	65.6	82.1	81.7	-	<b>84.8</b>
Waveform	1	57.8	74.3	66.9	<b>88.9</b>	73.3
	3	71.0	79.0	<b>86.7</b>	-	86.4
	5	69.3	85.6	<b>89.2</b>	-	<b>89.2</b>
	10	72.2	88.5	89.4	-	<b>89.6</b>
	15	83.5	89.0	<b>89.7</b>	-	89.4

### 4.3 Studies on Multi-class Datasets

The kernel EDA can be used in multiclass classification problem by converting the multiclass problem to multiple binary classification problems by using ‘‘One vs Rest’’ or ‘‘One vs One’’ scheme. The datasets of Vogel [12] and MIT [8] scene multi-class datasets used in our studies are given in Table 3. Local block features are used. Each image is divided into fixed size blocks. From each block, the color, edge direction histogram, and texture features are extracted. Thus, each block of image is represented by a 23-dimensional feature vector. The feature vectors from all the blocks in an image are concatenated to get the representation for an image.

The classification accuracies obtained on the transformed representation for the Vogel and MIT datasets using the linear SVM as classifier is given in Table 4. The kernel EDA technique gives a much higher accuracy as compared to the other techniques on MIT dataset. On Vogel dataset, all the techniques give a similar performance.

**Table 3.** Details of multi-class benchmark datasets used.

Name	No. of Examples	Dimension	No. of Classes
Vogel-6	700	2300	6
MIT-8	2688	828	8

**Table 4.** Classification Accuracies (in %) obtained on Vogel and MIT datasets for different methods for dimension reduction and for different values of reduced dimension. Performance is also compared for the one-versus-one (1-vs-1) and the one-vs-rest (1-vs-R) approach to multi class classification.

Dataset	Classification Approach	Reduced dimension ( $l$ )	PCA	Kernel PCA	Kernel ECA	Kernel FDA	GDA	Kernel EDA
MIT	1 vs 1	1	28.1	20.9	25.7	32.2	25.7	<b>38.9</b>
		5	42.9	37.2	41.3	-	46.3	<b>49.5</b>
		10	46.6	40.3	45.1	-	-	<b>50.9</b>
		20	47.8	45.6	47.2	-	-	<b>52.5</b>
		40	47.4	46.6	49.5	-	-	<b>53.4</b>
	1 vs R	1	19.4	9.6	<b>22.2</b>	16.5	16.5	21.8
		5	33.1	25.0	31.0	-	<b>43.9</b>	40.7
		10	39.3	35.7	39.9	-	-	<b>43.6</b>
		20	41.7	39.8	44.1	-	-	<b>46.3</b>
		40	46.5	42.1	45.6	-	-	<b>50.3</b>
Vogel	1 vs 1	1	30.3	29.6	32.9	25.0	24.3	<b>34.9</b>
		5	40.1	32.2	39.5	-	38.1	<b>42.8</b>
		10	<b>40.1</b>	39.5	38.8	-	-	<b>40.1</b>
		20	40.1	39.5	<b>43.2</b>	-	-	40.8
		40	40.1	<b>42.8</b>	<b>42.8</b>	-	-	40.8
	1 vs R	1	6.6	11.8	15.8	<b>16.4</b>	16.5	10.5
		5	30.9	29.6	37.5	-	<b>40.1</b>	35.5
		10	40.1	38.2	38.8	-	-	<b>45.4</b>
		20	<b>47.4</b>	39.5	40.1	-	-	42.1
		40	<b>49.3</b>	39.5	42.1	-	-	45.4

## 5 Conclusion

In this paper, we have proposed kernel entropy discriminant analysis as a data transformation method. It uses Euclidean divergence between the estimates of probability density functions of the two classes as the criterion function to decide the directions for projection. Though the kernel EDA is a supervised technique, it is not limited by the number of classes. Studies on various datasets show that proposed kernel EDA performs better or on-par as compared to PCA, kernel PCA, kernel ECA, kernel FDA and GDA.

## References

1. Baudat, G., Anouar, F.: Generalized discriminant analysis using a kernel approach. *Neural Comput.* **12**(10), 2385–2404 (2000)
2. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*. Wiley-Interscience, Hoboken (2000)
3. Fisher, R.A.: The use of multiple measurements in taxonomic problems. *Ann. Eugenics* **7**(2), 179–188 (1936)
4. Hotelling, H.: Relations between two sets of variates. *Biometrika* **28**(3/4), 321–377 (1936)
5. Izquierdo-Verdiguier, E., Laparra, V., Jenssen, R., Gmez-Chova, L., Camps-Valls, G.: Optimized kernel entropy components. *IEEE Trans. Neural Netw. Learn. Syst.* **28**(6), 1466–1472 (2017)
6. Jenssen, R.: Kernel entropy component analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(5), 847–860 (2010)
7. Mika, S., Ratsch, G., Weston, J., Scholkopf, B., Mullers, K.R.: Fisher discriminant analysis with kernels. In: *Proceedings of the 1999 IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing IX*, pp. 41–48 (1999)
8. Oliva, A., Torralba, A.: Modeling the shape of the scene: a holistic representation of the spatial envelope. *Int. J. Comput. Vision* **42**(3), 145–175 (2001)
9. Parzen, E.: On estimation of a probability density function and mode. *Ann. Math. Stat.* **33**(3), 1065–1076 (1962)
10. Principe, J.: *Information Theoretic Learning: Renyi’s Entropy and Kernel Perspectives*, 1st edn. Springer, New York (2010)
11. Schölkopf, B., Smola, A., Müller, K.-R.: Kernel principal component analysis. In: Gerstner, W., Germond, A., Hasler, M., Nicoud, J.-D. (eds.) *ICANN 1997. LNCS*, vol. 1327, pp. 583–588. Springer, Heidelberg (1997). doi:[10.1007/BFb0020217](https://doi.org/10.1007/BFb0020217)
12. Vogel, J., Schiele, B.: Semantic modeling of natural scenes for content-based image retrieval. *Int. J. Comput. Vision* **72**(2), 133–157 (2007)