# Kernel Principle Component Analysis in Pixels Clustering

Jing LI[1], Dacheng TAO[2], Weiming HU[3], and Xuelong LI[2]
[1]*Department of Electronic Engineering and Information Systems, Nanchang University*
[2]*School of Computer Science and Information Systems, Birkbeck College, University of London*
[3]*National Lab of Pattern Recognition, Institute of Automation, Chinese Academic Sciences*
*j.li@ncu.edu.cn; {dacheng, xuelong}@dcs.bbk.ac.uk; wmhu@nlpr.ia.ac.cn*

## Abstract

*We propose two new methods in the nonlinear kernel feature space for pixel clustering based on the traditional KMeans and Gaussian Mixture Model (GMM). Unlike the previous work on the kernel machines, we give out a new perspective on the new developed kernel machines. That is, kernel principle component analysis (KPCA) combined with the KMeans and the GMM are kernel KMeans (KKMeans) and kernel GMM (KGMM), respectively. In this paper, we prove the new perspective on KKMeans and give out a clear statement on the KGMM as well. Based on this new perspectives, we can implement the KKMeans and the KGMM conveniently. At the end of the paper, we utilize these new algorithms on the problem of the colour image segmentation. Based on a series of experimental results on Corel Colour Images, we find that the KKMeans and KGMM can outperform the traditional KMeans and GMM consistently, respectively.*

## 1. Introduction

Image segmentation [1] is useful in many applications for identifying regions of interest in a scene or annotating data. For example, the object-based image retrieval [2], the object tracking in video content analysis, general image content analysis and understanding, etc. Moreover, the MPEG-4 standard [3] needs segmentation for objection-based coding [3]. However, the problem of unsupervised segmentation is still an open problem in the field of the multimedia information processing.

Some of recent works on unsupervised segmentation include stochastic model-based approaches [4] [5] [6], the morphological watershed-based region growing [7], the energy diffusion [8], and graph cuts [9]. In this paper, we focus on the traditional unsupervised learning for image segmentation, such as the KMeans [10] algorithm and the Gaussian mixture model (GMM) [10] based segmentation. The motivation is because of the successes of the kernel machine [10] in machine learning both on the supervised learning and the unsupervised learning. In this paper, we give out a new perspective on the unsupervised kernel machine. That is, the kernel principle component analysis (KPCA) combined with the KMeans is the KMeans in the nonlinear kernel feature space (KKMeans). Similar to KKMeans, the KPCA incorporated with the GMM is the kernel GMM (KGMM). With these two new perspectives, we can implement the KKMeans and KGMM easily. Finally, we utilize the two new unsupervised learning perspectives on colour image segmentation.

## 2. Kernel KMeans Procedure

Clustering has received a significant amount of attention in the last few years as one of the fundamental problems in many application fields, such as the image segmentation, image database organization, data mining, etc. KMeans is one of the most popular clustering algorithms. Although, Dhillon et. al. developed the kernel KMeans, they do not give out a clear statement for KKMeans. In this paper, we proved that KPCA combined with the KMeans is the KKMeans in the theorem 1. Based on the theorem, we can understand the KKMeans much more clearly than the previous one.

***Theorem-1***: KPCA + KMeans is the kernel KMeans.

*Proof*:
To simplify the formulation, we assume that the data are zero centralized. The eigenvectors calculated in KPCA are denoted as:

$$\mathrm{P} = \left[ \beta^1, \beta^2, ..., \beta^N \right] \in R^{N \times N} .$$

Then,

$$K\beta^j = \lambda_j \beta^j ,\ j = 1, ..., N ,$$

For a data $z$, its projection to the $j^{\text{th}}$ projection in the higher dimensional Hilbert space is:

$$\left( \sum_i \beta_i^j \phi(x_i) \right)^T \phi(z) = \sum_i \beta_i^j k(x_i, z)$$

Denote $X_{KPCA}$ as the projections of all the samples to all computed KPCA projection directions. Then, we conduct the key step (calculating the distance between a given sample and a given centroid) of KMeans on the projected datum $a_{KPCA}$ as:

Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)
0-7695-2415-X/05 $20.00 © 2005 IEEE

$$\left\|a_{KPCA}-m_{KPCA|j}\right\|^2=\left(a_{KPCA}-m_{KPCA|j}\right)^T\left(a_{KPCA}-m_{KPCA|j}\right)$$
$$=a_{KPCA}^T a_{KPCA}-2a_{KPCA}^T m_{KPCA|j}+m_{KPCA|j}^T m_{KPCA|j}$$
$$=\begin{pmatrix} a_{KPCA}^T a_{KPCA}-2a_{KPCA}^T\sum_{b\in\pi_j} b_{KPCA} \\ +\sum_{b\in\pi_j} b_{KPCA}^T\sum_{c\in\pi_j} c_{KPCA}\end{pmatrix}$$

Then we have:

$$=\begin{bmatrix}\sum_i\beta_i^1 k(x_i,a)\\ \dots\\ \sum_i\beta_i^N k(x_i,a)\end{bmatrix}^T\begin{bmatrix}\sum_i\beta_i^1 k(x_i,a)\\ \dots\\ \sum_i\beta_i^N k(x_i,a)\end{bmatrix}$$
$$-\frac{2}{n_j}\sum_{b\in\pi_j}\begin{bmatrix}\sum_i\beta_i^1 k(x_i,a)\\ \dots\\ \sum_i\beta_i^N k(x_i,a)\end{bmatrix}^T\begin{bmatrix}\sum_i\beta_i^1 k(x_i,b)\\ \dots\\ \sum_i\beta_i^N k(x_i,b)\end{bmatrix}$$
$$+\frac{1}{n_j^2}\sum_{b\in\pi_j}\sum_{c\in\pi_j}\begin{bmatrix}\sum_i\beta_i^1 k(x_i,b)\\ \dots\\ \sum_i\beta_i^N k(x_i,b)\end{bmatrix}^T\begin{bmatrix}\sum_i\beta_i^1 k(x_i,c)\\ \dots\\ \sum_i\beta_i^N k(x_i,c)\end{bmatrix}$$

According to the kernel trick:

$$=\begin{bmatrix}\left(\sum_i\beta_i^1\phi(x_i)\right)^T\phi(a)\\ \dots\\ \left(\sum_i\beta_i^N\phi(x_i)\right)^T\phi(a)\end{bmatrix}^T\begin{bmatrix}\left(\sum_i\beta_i^1\phi(x_i)\right)^T\phi(a)\\ \dots\\ \left(\sum_i\beta_i^N\phi(x_i)\right)^T\phi(a)\end{bmatrix}$$
$$-\frac{2}{n_j}\sum_{b\in\pi_j}\begin{bmatrix}\left(\sum_i\beta_i^1\phi(x_i)\right)^T\phi(a)\\ \dots\\ \left(\sum_i\beta_i^N\phi(x_i)\right)^T\phi(a)\end{bmatrix}^T\begin{bmatrix}\left(\sum_i\beta_i^1\phi(x_i)\right)^T\phi(b)\\ \dots\\ \left(\sum_i\beta_i^N\phi(x_i)\right)^T\phi(b)\end{bmatrix}$$
$$+\frac{1}{n_j^2}\sum_{b\in\pi_j}\sum_{c\in\pi_j}\begin{bmatrix}\left(\sum_i\beta_i^1\phi(x_i)\right)^T\phi(b)\\ \dots\\ \left(\sum_i\beta_i^N\phi(x_i)\right)^T\phi(b)\end{bmatrix}^T\begin{bmatrix}\left(\sum_i\beta_i^1\phi(x_i)\right)^T\phi(c)\\ \dots\\ \left(\sum_i\beta_i^N\phi(x_i)\right)^T\phi(c)\end{bmatrix}$$

Finally, we get:

$$=\begin{pmatrix}\left(P^T\left[\phi(x_1)\ \dots\ \phi(x_N)\right]^T\phi(a)\right)^T\\ \left(P^T\left[\phi(x_1)\ \dots\ \phi(x_N)\right]^T\phi(a)\right)\end{pmatrix}$$
$$-\frac{2}{n_j}\sum_{b\in\pi_j}\begin{pmatrix}\left(P^T\left[\phi(x_1)\ \dots\ \phi(x_N)\right]^T\phi(a)\right)^T\\ \left(P^T\left[\phi(x_1)\ \dots\ \phi(x_N)\right]^T\phi(b)\right)\end{pmatrix}$$
$$+\frac{1}{n_j^2}\sum_{b\in\pi_j}\sum_{c\in\pi_j}\begin{pmatrix}\left(P^T\left[\phi(x_1)\ \dots\ \phi(x_N)\right]^T\phi(b)\right)^T\\ \left(P^T\left[\phi(x_1)\ \dots\ \phi(x_N)\right]^T\phi(c)\right)\end{pmatrix}$$

That is:

$$=\begin{pmatrix}\left(\phi^T(a)\left[\phi(x_1)\ \dots\ \phi(x_N)\right]P\right)\\ \left(P^T\left[\phi(x_1)\ \dots\ \phi(x_N)\right]^T\phi(a)\right)\end{pmatrix}$$
$$-\frac{2}{n_j}\sum_{b\in\pi_j}\begin{pmatrix}\left(\phi^T(a)\left[\phi(x_1)\ \dots\ \phi(x_N)\right]P\right)\\ \left(P^T\left[\phi(x_1)\ \dots\ \phi(x_N)\right]^T\phi(b)\right)\end{pmatrix}$$
$$+\frac{1}{n_j^2}\sum_{b\in\pi_j}\sum_{c\in\pi_j}\begin{pmatrix}\left(\phi^T(b)\left[\phi(x_1)\ \dots\ \phi(x_N)\right]P\right)\\ \left(P^T\left[\phi(x_1)\ \dots\ \phi(x_N)\right]^T\phi(c)\right)\end{pmatrix}$$

Because of the normalized orthogonal property of the KPCA, we have

$$\begin{pmatrix}\left(\left[\phi(x_1)\ \dots\ \phi(x_N)\right]P\right)\\ \left(\left[\phi(x_1)\ \dots\ \phi(x_N)\right]P\right)^T\end{pmatrix}=0$$

Consequently, we can simplify the deduction as:

$$\begin{pmatrix}\phi^T(a)\phi(a)-\frac{2}{n_j}\sum_{b\in\pi_j}\phi^T(a)\phi(b)\\ +\frac{1}{n_j^2}\sum_{b\in\pi_j}\sum_{c\in\pi_j}\phi^T(b)\phi(c)\end{pmatrix}$$

With the formulation, we have the following equation:

$$\left\|a_{KPCA}-m_{KPCA|j}\right\|^2=\left\|\phi(a)-\frac{1}{n_j}\sum_{b\in\pi_j}\phi(b)\right\|^2$$

Based on the above analysis, we can draw the conclusion the conclusion that the distance between a given point and a given centroid in the KPCA space equals to the distance in the kernel space. Therefore, KPCA plus the KMeans is the Kernel KMeans. ■

With the theorem, we know that the implementation of KKMeans is simple. That is we can do the KPCA as the preprocessing step and then conduct the KMeans on the KPCA space.

## 3. Gaussian Mixture Model in the Nonlinear Kernel Feature Space

Another important model for unsupervised learning is the GMM. GMM is a generalized type of KMeans, which measure the divergence by the Mahalanobis distance on the probability space. Before we give out the procedure for KGMM, we first revisit the GMM. The Gaussian Mixture Model is:

$$p(x\mid\Theta)=\sum_{i=1}^{M}\alpha_i p_i(x\mid\theta_i)$$

where the parameters are $\Theta=(\alpha_i,\theta_i\mid 1\le i\le M)$ such that $\sum_{i=1}^{M}\alpha_i=1$ and each $p_i$ is a Gaussian density function parameterized by $\theta_i$ as:

$$\begin{aligned} p_i(x\mid\theta_i) &= p_i(x\mid\mu_i,\Sigma_i) \\ &= \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}}\exp\left\{-\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)\right\}. \end{aligned}$$

Generally, the expectation maximization (EM) procedure is conducted iteratively to estimate the parameters. The EM procedure estimates the updated parameters in terms of the old parameters are given as follows:

$$\alpha_i^{new} = \frac{1}{N}\sum_{l=1}^{N} p(i\mid x_l,\Theta^g)$$

$$\mu_i^{new} = \frac{\sum_{l=1}^{N} x_l\, p(i\mid x_l,\Theta^g)}{\sum_{l=1}^{N} p(i\mid x_l,\Theta^g)}$$

$$\Sigma_i^{new} = \frac{\sum_{l=1}^{N} p(i\mid x_l,\Theta^g)(x_l-\mu_i^{new})(x_l-\mu_i^{new})^T}{\sum_{l=1}^{N} p(i\mid x_l,\Theta^g)}$$

where $p(i\mid x_l,\Theta^g) = \dfrac{\alpha_i^g\, p(x_l\mid\theta_i^g)}{\sum_{j=1}^{M}\alpha_j^g\, p(x_l\mid\theta_j^g)}$.

Note that the above equations perform both the expectation step and the maximization step simultaneously. The algorithm proceeds by using the newly derived parameters as the guess for the next iteration.

GMM cannot work well on the nonlinear feature space. Therefore, we generalize the GMM to the kernel space. With the GMM and the kernel mapping, we can obtain the KGMM conveniently. The Gaussian Mixture Model in the nonlinear kernel feature space is:

$$p(\phi(x)\mid\Theta) = \sum_{i=1}^{M}\alpha_i p_i(\phi(x)\mid\theta_i)$$

where the parameters are $\Theta = (\alpha_i,\theta_i \mid 1\le i\le M)$ such that $\sum_{i=1}^{M}\alpha_i = 1$ and each $p_i$ is a Gaussian density function parameterized by $\theta_i$ in the kernel space as:

$$\begin{aligned} p_i(\phi(x)\mid\theta_i) &= p_i(\phi(x)\mid\mu_i,\Sigma_i) \\ &= \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}}\exp\left\{-\frac{1}{2}(\phi(x)-\mu_i)^T\Sigma_i^{-1}(\phi(x)-\mu_i)\right\}. \end{aligned}$$

According to the formulation and the kernel mapping theory, we know that the dimension of $\phi(x)$ is generally much higher than the number of the training samples. In our problem, the training set and the testing set are same, thus the null subspace (all the samples are mapped onto the same point on the subspace) of the covariance matrix of all data points will not be useful for clustering. Therefore, we can remove the null subspace before learning by conducting the principle component analysis on the kernel space and preserve the principle subspace for learning.

Here, we use the EM procedure to estimate the parameters and the update formula are given by:

$$\alpha_i^{new} = \frac{1}{N}\sum_{l=1}^{N} p(i\mid\phi(x_l),\Theta^g) = \frac{1}{N}\sum_{l=1}^{N} p(i\mid x_l^{KPCA},\Theta^g)$$

$$\mu_i^{new} = \frac{\sum_{l=1}^{N} x_l\, p(i\mid\phi(x_l),\Theta^g)}{\sum_{l=1}^{N} p(i\mid\phi(x_l),\Theta^g)} = \frac{\sum_{l=1}^{N} x_l\, p(i\mid x_l^{KPCA},\Theta^g)}{\sum_{l=1}^{N} p(i\mid x_l^{KPCA},\Theta^g)}$$

$$\begin{aligned} \Sigma_i^{new} &= \frac{\sum_{l=1}^{N} p(i\mid\phi(x_l),\Theta^g)(\phi(x_l)-\mu_i^{new})(\phi(x_l)-\mu_i^{new})^T}{\sum_{l=1}^{N} p(i\mid\phi(x_l),\Theta^g)} \\ &= \frac{\sum_{l=1}^{N} p(i\mid x_l^{KPCA},\Theta^g)(x_l^{KPCA}-\mu_i^{new})(x_l^{KPCA}-\mu_i^{new})^T}{\sum_{l=1}^{N} p(i\mid x_l^{KPCA},\Theta^g)} \end{aligned}$$

where

$$\begin{aligned} p(i\mid\phi(x_l),\Theta^g) &= \frac{\alpha_i^g\, p(\phi(x_l)\mid\theta_i^g)}{\sum_{j=1}^{M}\alpha_j^g\, p(\phi(x_l)\mid\theta_j^g)} \\ &= \frac{\alpha_i^g\, p(x_l^{KPCA}\mid\theta_i^g)}{\sum_{j=1}^{M}\alpha_j^g\, p(x_l^{KPCA}\mid\theta_j^g)} = p(i\mid x_l^{KPCA},\Theta^g) \end{aligned}.$$

Moreover,

$$\begin{aligned} p_i(\phi(x)\mid\theta_i) &= p_i(\phi(x)\mid\mu_i,\Sigma_i) \\ &= \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}}\exp\left\{-\frac{1}{2}(\phi(x)-\mu_i)^T\Sigma_i^{-1}(\phi(x)-\mu_i)\right\} \\ &= \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}}\exp\left\{-\frac{1}{2}(x^{KPCA}-\mu_i)^T\Sigma_i^{-1}(x^{KPCA}-\mu_i)\right\} \\ &= p_i(x^{KPCA}\mid\mu_i,\Sigma_i) = p_i(x^{KPCA}\mid\theta_i) \end{aligned}$$

and the GMM in kernel space is given by:

$$\begin{aligned} p(\phi(x)\mid\Theta) &= \sum_{i=1}^{M}\alpha_i p_i(\phi(x)\mid\theta_i) \\ &= \sum_{i=1}^{M}\alpha_i p_i(x^{KPCA}\mid\theta_i) = p(x^{KPCA}\mid\Theta) \end{aligned}$$

From the deduction of the GMM in the nonlinear kernel feature space, we can easily prove that KPCA combined with the GMM is the kernel GMM. The proof procedure is similar to the proof of Theorem 1.

## 4. Colour Image Segmentation

In colour image segmentation, there are three types of information that can contribute the segmentation: the Luv colour components [11], the pixel position information, and the Gabor texture [13]. Some preliminary image segmentation results are given in the following Figs. The 1st row is the original images, the 2nd is the kmeans, the 3rd is kkmeans, the 4th is the GMM, and the final is the KGMM.

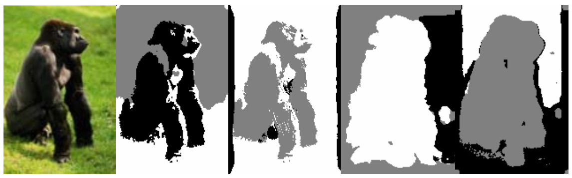

**Figure 1. Segmentation results.**

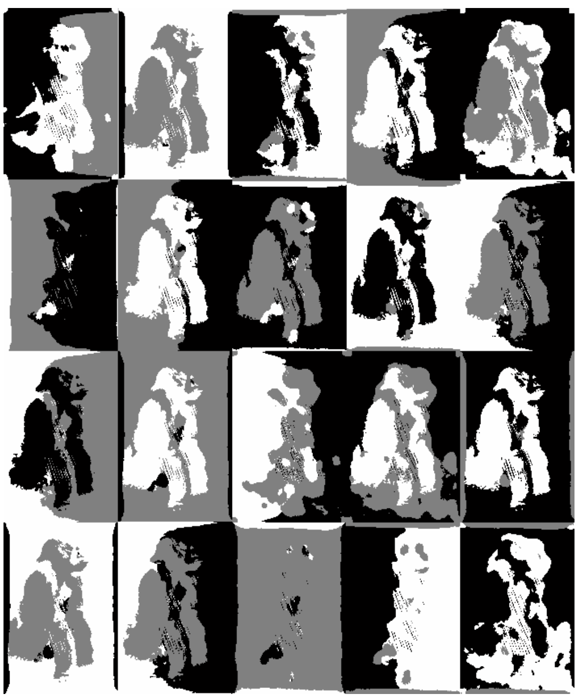

**Figure 2. Image Segmentation results in different kernel parameter in KKMeans.**

## 5. Conclusion

In this paper, we give out and prove a novel perspective on some kernel machines. That is, kernel principle component analysis (KPCA) combined with the KMeans and the GMM are kernel KMeans (KKMeans) and kernel GMM (KGMM), respectively. Through the new perspectives, we can implement the KKMeans and the KGMM conveniently.

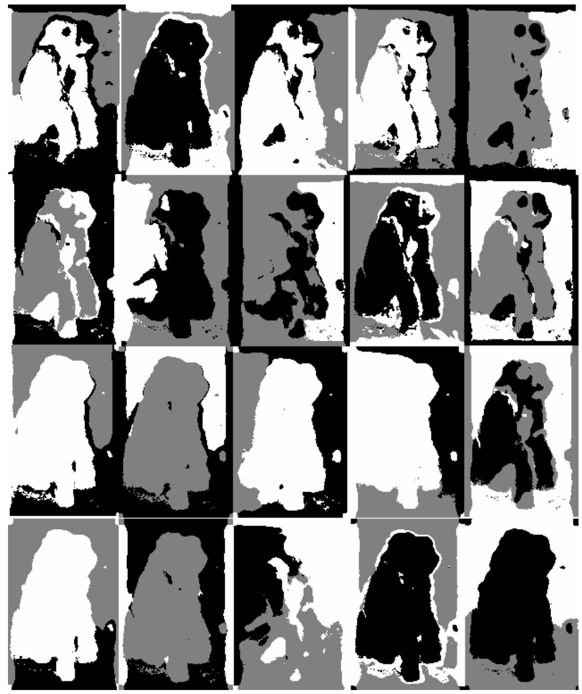

**Figure 3. Image Segmentation results in different kernel parameter in KGMM.**

## References


[1] Y. Deng and B. S. Manjunath, "Unsupervised Segmentation of Color-Texture Regions in Images and Video," IEEE TPAMI, 2001.

[2] D. Tao, J. Liu, and X. Tang, "Learning User's Perception Using Region-based SVM for Content-based Image Retrieval," CISST 2004.

[3] http://www.wave-report.com/tutorials/VC.htm

[4] Y. Delignon, A. Marzouki, and W. Pieczynski, "Estimation of Generalized Mixtures and Its Application in Image Segmentation," IEEE Trans. Image Proc. 6(10): 1364-1376 (1997).

[5] D. A. Langan, J. W. Modestino, and J. Zhang, "Clustering Validation for Unsupervised Stochastic Model-Based Image Segmentation," IEEE Trans. Image Proc. 7(2):180-244 (1997).

[6] J. Z. Wang, J. Li, R. M. Gray, and G. Wiederhold, "Unsupervised Multiresolution Segmentation for Images with Low Depth of Field," IEEE Trans. Pattern Anal. Mach. Intell. 23(1): 85-90 (2001).

[7] K. Haris, S.N. Efstratiadis, and N. Maglaveras, "Watershed-based image segmentation with fast region merging," ICIP, 1998.

[8] W. Y. Ma and B. S. Manjunath, "Edge Flow: a Framework of Boundary Detection and Image Segmentation," CVPR, 1997.

[9] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," IEEE Trans. Pattern Anal. Mach. Intell. 22(8): 888-905 (2000).

[10] R. O. Duda, P. E. Hart and D. G. Stork, "Pattern Classification," 2nd Ed. Wiley 2001.

[11] http://www.neuro.sfc.keio.ac.jp/~aly/polygon/info/color-space-faq.html

[12] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying," IEEE TPAMI, 2002.

[13] X. Wang and X. Tang, "Bayesian Face Recognition Using Gabor Features," In Proc. ACM Workshop on WBMA 2003.





IEEE COMPUTER SOCIETY