KD-Trees 구조를 이용한 MPEG 비디오 검색

김대일, 홍종선, 장혜경, 김영호, 강대성
동아대학교 전자공학과

MPEG Video Retrieval Using KD-Trees Construction

Daeil Kim, Jong-Sun Hong, Hye-Kyoung Jang, Young Ho Kim, Dae-Seong Kang
Department of Electronic Engineering, Dong-A University
E-mail : daeilu@empal.com

Abstract

In this paper, we propose image retrieval method more accurate and efficient than the conventional one. First of all, we perform a shot detection and key frame extraction from the DC image constructed by DCT DC coefficients in the compressed video stream that is video compression standard such as MPEG[1][2]. We get principal axis applying PCA(Principal Component Analysis) to key frames for obtaining indexing information, and divide a domain. Video retrieval uses indexing information of high dimension. We apply KD-Trees(K Dimensional-Trees)[3] which shows efficient retrieval in data set of high dimension to video retrieval method. The proposed method can represent property of images more efficiently and property of domains more accurately using KD-Trees.

I. Introduction

Image retrieval methods can divide widely into test-based retrieval and content-based retrieval. Early image database system is text-based retrieval, and this method retrieves as using the information after adding representative notes about each image. But if multimedia information represents with text-based information, it is supervised and has a fault that cannot retrieve image similar to query image. Content-based retrieval, on the contrary, there are merits that can extract feature of image and store that information automatically[4][5][6]. Multimedia data is now one of the widely information in all the fields as the fast developments of computer techniques have been made. There is a rapid increase in the use of digital video information in recent years, it becomes more important to manage multimedia database efficiently. There is a big concern about video indexing because users require content-based image retrieval.

In this paper, we propose image retrieval method using KD-Trees construction. The KD-Trees which is the one of many methods using the indexing information of high dimensional points is the algorithm that shows progressive retrieval. Coordinates of a data dimensions increases, the entry size becomes larger. As a result, the node capacity is reduced. Therefore, the search procedure has to follow longer paths to reach the leaf nodes, which decreases retrieval performance. This problem has effect on video retrieval because it handles the indexing information of high dimension. KD-Trees prevents this problem from occurring, can make it efficiently retrieve.

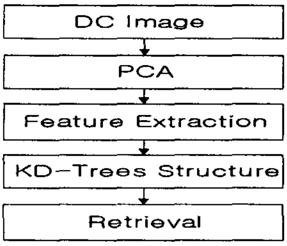


Figure 1. Total block diagram

Figure 1 is total block diagram in this paper. Experimental result shows that the proposed method outperforms the conventional one.

II. Extraction of DC image

MPEG video of international moving picture standard consists of picture of I, P, B type. In this paper, we make DC image after being normalized DCT DC coefficients extracted from I picture of MPEG video streams.

Figure 1. DC image from MPEG video stream
(a) I-type frame, (b) DC image

Figure 1(a) is to extract key frame of MPEG video stream and the size is 352×240 image and (b) is DC image constructed DCT DC coefficients of key frame, 44×30 image.

III. Indexing using PCA

PCA algorithm has a merit that can get good result if we select properly initial error and can generate codebook with high speed because of no training time against neural network. We solve principal axis using PCA algorithm and then we divide region using region-division algorithm.

1. Algorithm to solve principal axis

Step 1. Calculate geometrical mean, M about input vector

$$M = \begin{bmatrix} Mx \\ My \end{bmatrix}$$

$$M_x = \frac{1}{q}\sum_{i=0}^{q-1} Ix_i \ , \ My = \frac{1}{q}\sum_{i=0}^{q-1} Iy_i$$

Here, q is number of total input vector, and Ix_i and Iy_i is value of x axis and value of y axis of input vector.

Step 2. Calculate covariance matrix of image

$$C = \begin{bmatrix} C_1 & C_2 \\ C_3 & C_4 \end{bmatrix} \quad , \textit{where}$$

$$C_1 = \frac{1}{q}\sum_{i=0}^{q-1}\left(x_i^2 - M_x^2\right)$$

$$C_2 = C_3 = \frac{1}{q}\sum_{i=0}^{q-1}\left(x_i y_i - M_x M_y\right)$$

$$C_4 = \frac{1}{q}\sum_{i=0}^{q-1}\left(y_i^2 - M_y^2\right)$$

Step 3. Calculate the largest eigen-value

$$\lambda_{\max} = \frac{(C_1 + C_4) + \sqrt{(C_1 + C_4)^2 - 4(C_1C_4 - C_2C_3)}}{2}$$

Step 4. Calculate an angle of principal axis

$$\theta = a\tan\left[\frac{\lambda_{\max} - C_1}{2}\right]$$

After calculating angle of principal axis like this[7], the region is divided using KD-tree. Figure 2 is to reduce two dimensional attributes in data one dimension for image analysis.

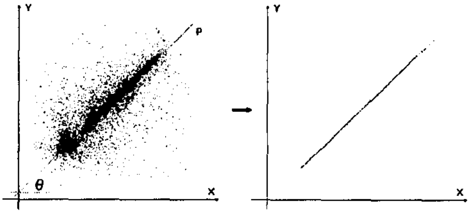


Figure 2. Reduction of a dimension by PCA

2. Region division algorithm

Step 1. Calculate average axis of input vector of same region

Step 2. Calculate angle(φ) with line passing average coordinates and vertical with principal axis using algorithm of principal-axis angle calculation

Step 3. If (cos(φ) $\leq$ 0) then region 1 else region 2

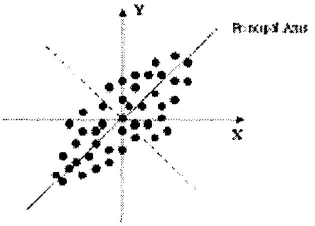


Figure 3. Region segmentation using principal axis

Figure 3 is to express figure of divided region after calculating principal axis by proposed algorithm.

IV. KD-Trees

KD-Trees is to eliminate the redundant information from the retrieval. Removal of redundant information increases the capacity of the index nodes. Figure 4 shows three index regions. In the KD-Trees, the index entry for region R1 is <1, 0, 0.4, cp1>, for region R2, <2, 0.4, 0.5, 1, 1, cp2>, and for region R3, <2, 0.4, 0, 1, 0.5, cp3>, where cp1, cp2, and cp3 are child-node pointers. Here a value of 0.4 is shared by all three indes entries, and a value of 0.5 is shared by two index entries.

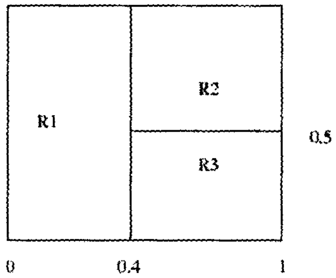


Figure 4. Three index regions

In the figure 5, to insert a new query object, the tree has to be traversed from the root to a leaf page. At each division node, the left branch is chosen if the node value is greater than a new query object. Otherwise, the right branch is chosen. If the query object reaches leaf nodes, retrieval procedure will finish.

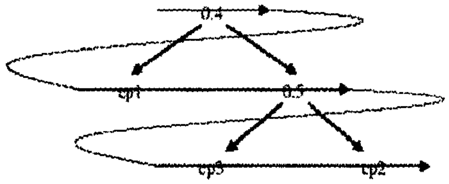


Figure 5 KD-Trees structure

Figure 6 shows KD-Trees structure expanded a concept in the video stream.

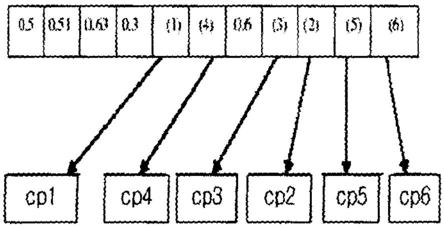
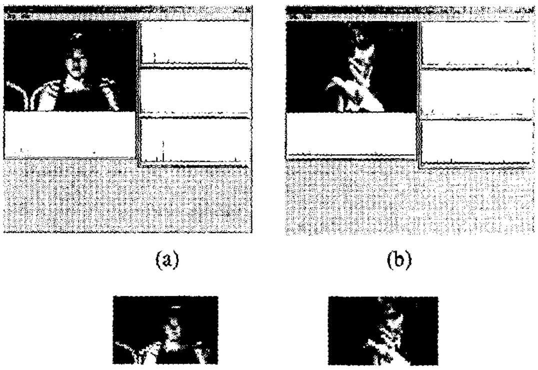


Figure 6. An example of the KD-Trees structure

V. Experimental Results

In this experiment, we make key frames extracted MPEG video stream and convert DC images into indexing information by PCA algorithm.



(a) (b)

(c) (d)

Figure 6. (a), (b) DC image extractor (c), (d) its result

Figure 6(a), (b) is DC image extractor on real-time and (c), (d) is the result images extracted from (a), (b). Figure 5 makes DC image about key frame of Figure 1 to two dimensional vector and results in 50, 100, 200, and 300 by final node number though tree-map construction.

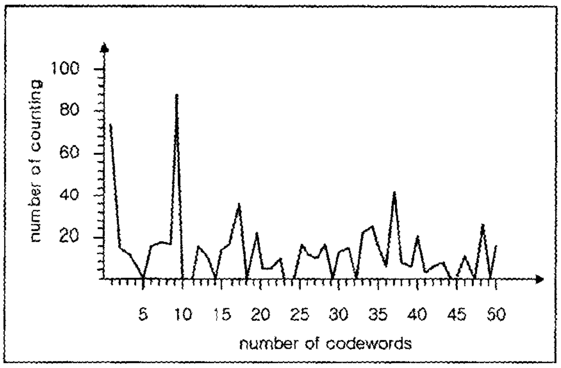
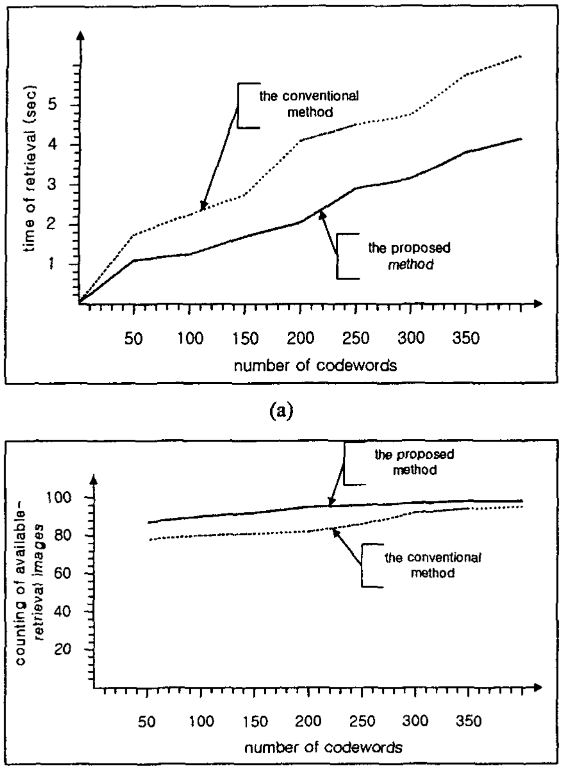


Figure 6. The result of codebook

Figure 6 is plots of codebook about figure 1. The width is to express 50 codewords of key vectors because of limited space and the length is to count each codewords. In the figure 7(a), we compare the time of the proposed method with the conventional method about each codewords. This shows that the time of the proposed method decreases more and more by contrast with the conventional method. In the figure 7(b), we compare the retrieval performance of two method. The proposed method outperform the conventional one.



(a)

(b)

Figure 7. (a) The time of , (b) Performance of the proposed method and the conventional method

Figure 8 is the retrieval system used KD-Trees algorithm. It shows that the system placed images retrieved accurately in order about the query image.

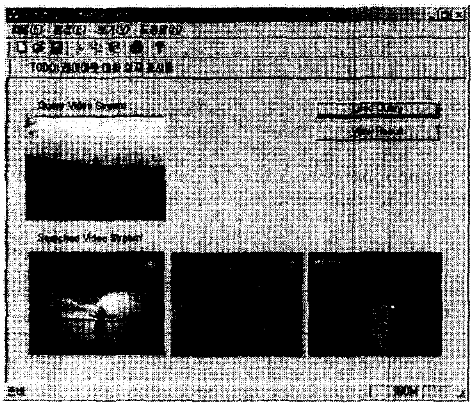


Figure 8. Retrieval system used the proposed method

VI. Conclusions

In this paper, we propose a KD-Trees algorithm that performs well in MPEG video retrieval. we apply KD-Trees algorithm to MPEG video retrieval using PCA. The tree-map makes leaf nodes from each DC image. The proposed method can represent property of images more efficiently using and property of domains more accurately using KD-Trees. Therefore, we can reduce the computational expense and memory usage, and can obtain high accuracy more than conventional methods did. Henceforward, we will want to decrease page capacity of retrieval procedure using a study on split and merge in the KD-Trees structure.

References

[1] Wei Xiong and Chung-Mong Lee, "Efficient Scene Change Detection and Camera Motion Annotation for Video Classification", Computer Vision and Image Understanding Vol.71, No. 2/2, August, pp. 166-181, 1998.

[2] Hong Heather Yu, "A hierarchical Multiresolution Video shot Transition Detection Scheme , Computer Vision and Image Understanding Vol.75, No. 1/2, July/ August, pp. 196-213, 1999.

[3] J. L. Bently, "Multidimensional binary search trees used for associative searching", Communications of ACM 18(9), pp. 509-517, 1975.

[4] Markus Stricker and Alexander Dimai, "Colir Indesing with Weak Spatial Constraint", Storage and Retrieval for Image and Video Database IV, SPIE proceedings, Vol. 2670, pp. 29-40, 1996.

[5] I. K. Park, I. D. Yun and S. U. Lee, "Color Image Retrieving using Hybrid Graph Representation", Image and Vision Computing, Vol. 17, No. 7, pp. 465-474, 1999.

[6] D. Androutsos, K. N. Plataniotia and A. N. Venetsanopoulos, "Distance Measures for Color Image Retrieval", International Conference on Image Processing, Vol. 2, pp. 770-774, 1998.

[7] Oja, E., "A simplified neuron model as a principal component analyzer,"Journal of Mathematical Biology 15, pp. 267-273, 1982