IMAGE RETRIEVAL AND CLASSIFICATION USING AFFINE INVARIANT B-SPLINE REPRESENTATION AND NEURAL NETWORKS*

Yiannis Xirouhakis, Yannis Avrithis and Stefanos Kollias

Department of Electrical and Computer Engineering
National Technical University of Athens
Heroon Polytechneiou 9, 157 73 Zographou, Greece
e-mail: jxiro@image.ntua.gr

ABSTRACT

In this paper, a system for content-based image retrieval from video databases is introduced, using B-splines for affine invariant object representation. A small number of "key-frames" is extracted from each video sequence, which provide sufficient information about the video content. Color and motion segmentation and tracking is then employed for automatic extraction of video objects. A B-spline representation of the object contours is then obtained, which possesses important properties, such as smoothness, continuity and invariance under affine transformation. A neural network approach is used for supervised classification of video objects into prototype object classes. Finally, higher level classes can be constructed combining primary classes, providing the ability to obtain a high level of abstraction in the representation of each video sequence.

1. INTRODUCTION

Due to recent growth in interest in multimedia applications, an increasing demand has emerged for efficient storage, management and browsing in multimedia databases. The latter has been given considerable attention after the recent guidelines of the Moving Pictures Expert Group regarding the MPEG-4 and MPEG-7 standards. Content-based query, retrieval and indexing capabilities are of paramount importance in browsing digital video databases, due to the vast amount of information involved.

Some prototype systems which provide such capabilities, including Photobook, Virage, VisualSEEk and QBIC have already been developed and are now in the stage of validation. These systems enable searching through on-line image databases and still image retrieval through a web interface using color, texture and shape attributes. Moreover, several works have been proposed in recent literature for the extension of the aforementioned schemes to video databases. These include video object modeling and segmentation [1], semantically meaningful feature spaces [2], and optimal extraction of frames and scenes [3]. Some prototype systems have also been proposed, giving the ability of querying-by-sketch in image databases, using image curvelet feature extraction and matching [4] and B-splines [5].

Based on the above work, a new system is introduced in this paper, extending the use of B-spline object contour representation to video queries and allowing video object matching and classification based on object shape apart from other features (such as color, texture, motion etc.). Furthermore the object representation obtained in this paper can be generalised in order to achieve video content description with a high level of abstraction.

The proposed system consists of several blocks. Initially, each video sequence of the video database is partitioned into video shots. An unsupervised color and motion segmentation technique is then applied to all frames of each video shot, and segment characteristics are used to construct a feature vector for each frame. Using an optimization method for locating a set of minimally correlated feature vectors, a small number of key frames and shots is selected so that subsequent processing is constrained to only a subset of the original sequence. All of the above procedures are described in detail in [3].

Object contours are obtained from image segments and a B-spline representation is used to model the resulting curves. A neural network approach is used for supervised classification of video objects into prototype object classes, which are used for the construction of an object class database. The classification scheme includes user interaction with the database, in order to perform queries and potentially update the stored prototypes. The above mentioned modules are further described in the sequel.

2. B-SPLINE REPRESENTATION

Using the object contours obtained through segmentation of the key-frames, a curve modeling scheme should be applied in order to facilitate recognizing and matching object shapes. A number of different approaches have been proposed for this purpose such as B-splines, Fourier descriptors, chain codes etc. In this work B-splines are employed since they possess a number of properties which make them suitable for shape representation and analysis such as smoothness and continuity, built-in boundedness, local controlability and shape invariance under affine transformation. In addition Fourier descriptors and curve moments are utilized for quick curve classification and analytical affine-parameter estimation respectively, as it will be seen in the sequel.

Curve Modelling. Assume that we are given a dense set of m data curve points $\mathbf{s}_j$, $j = 0,..,m\text{-}1$. The initial goal is to model the input curve using closed cubic B-splines that consist of $n+1$ connected curve segments $\mathbf{r}_i$, $i = 0,1,..,n$. Each of these segments is a linear combination of four cubic polynomials in the parameter $t \in [0,1]$:

$$\mathbf{r}_i(t) = \mathbf{C}_{i-1}Q_0(t) + \mathbf{C}_i Q_1(t) + \mathbf{C}_{i+1}Q_2(t) + \mathbf{C}_{i+2}Q_3(t)$$

* This work was funded by the National Program "YPER" by the General Secretariat of Research & Development of Greece entitled "Efficient Content-Based Image and Video Query and Retrieval in Multimedia Systems"

for $i = 0,1,..,n$, where $Q_k(t) = a_{k0}t^3 + a_{k1}t^2 + a_{k2}t + a_{k3}$, $k = 0,1,2,3$.

Using the continuity constraints in position, slope and curvature on the connection points between segments and the invariance property to coordinate transformations ($\sum_{k=0}^{3} Q_k(t) = 1,\ t \in [0,1]$), the polynomial factors a_k are computed and thus the basis functions $Q_k(t)$ are defined. The B-spline used to model the input curve is given using the curve segments as:

$$\mathbf{r}(t') = \sum_{k=0}^{n} \mathbf{r}_i (t' - i) = \sum_{k=0}^{n} \mathbf{C}_{i \bmod(n+1)} N_i(t')$$

where $0 \le t' \le n-2$ and $N_i(t)$ denote the so-called blending functions:

$$N_i(t') = \begin{cases} Q_3(t' - i + 3) & i-3 \le t' < i-2 \\ Q_2(t' - i + 2) & i-2 \le t' < i-1 \\ Q_1(t' - i + 1) & i-1 \le t' < i \\ Q_0(t' - i) & i \le t' < i+1 \\ 0 & otherwise \end{cases}$$

In order to find the appropriate B-spline, the control points $\mathbf{C}_i$ must be determined. The approach followed in this work tries to find an approximate B-spline such that the error between the observed data and their corresponding B-spline curve is minimized. In this sense, the metric $d^2 = \sum_{j=1}^{m} \left\| \mathbf{s}_j - \mathbf{r}(t'_j) \right\|^2$ should be minimized. If appropriate parametric values of t' are allocated on the curve, then the MMSE solution for the control points is given in matrix form as $\mathbf{C}_f = (\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{f}$, where $\mathbf{f}$ and $\mathbf{C}_f$ are of size $m \times 2$ and $(n+1) \times 2$ respectively containing the given data points $\mathbf{s}_j$ and the control points $\mathbf{C}_i$ respectively. The $m \times (n+1)$ matrix $\mathbf{P}$ contains appropriate values for the blending functions, estimated on the points $\mathbf{r}(t'_j)$, as shown in the equation at the bottom of the page.

For the allocation of parametric values of t', the chord length (CL) method is employed. Specifically, for $t'_1 = 0$ and $t'_{\max} = n-2$, t'_j associated with the sample point $\mathbf{s}_j$ is estimated by:

$$t'_j = t'_{j-1} + t'_{\max} \cdot \left\| \mathbf{s}_j - \mathbf{s}_{j-1} \right\| \cdot \left(\sum_{l=2}^{m} \left\| \mathbf{s}_l - \mathbf{s}_{l-1} \right\| \right)^{-1}, \quad j = 2,...,m \text{ T}$$

he CL is based on the fact that the chord length between any two points is a very close approximation to the arc length of the curve and under the assumption of constant speed of a particle onto the curve. The CL method is robust to uniformly distributed noise, but suffers from nonuniform noise and nonuniform sampling. Alternatively, the inverse chord length method (ICL) could be used for robust results, as reported in [7].

Curve Matching. In the sequel, the problem of comparing and matching curves using their B-spline representation is addressed. Assume that a set of *M* different curves, i.e. *M* sets of samples, are available in the database. After having modeled these sets of points with *M* cubic B-splines, it can be seen that their control points cannot decide shape similarity between these curves, since generally different sets of control points may describe the same curve.

For this reason, it is comfortable to derive foreach curve the so-called knot points $\mathbf{p}_i$, i=0,1,...,n, using the estimated control points. For cubic B-splines, this is achieved as $\mathbf{p}_f = \mathbf{A}\mathbf{C}_f$, where $\mathbf{p}_f$ is the $(n+1) \times 2$ matrix containing the knot-points and $\mathbf{A}$ is the $(n+1) \times (n+1)$ circulant matrix with [2/3,1/6,0,...,0,1/6] as its first row. It must be pointed out here that the knot-points belong to the derived B-spline.

However, it can be seen that for any two curves, it is not certain that their estimated knot-points correspond, even if they are equal in number. For this reason, they must be re-allocated on each curve [6]. The first knot-point is placed on the curve point where the curve intersects the x-axis. In the sequel, we place l knot-points equally spaced w.r.t. t' onto each curve. The underlying reason for this method is that for any input sample curve in the system, their re-allocated knot-points correspond always.

The classifier based on the re-allocated knot-points is based on minimizing a metric such as $d^2 = \sum_{i=1}^{l} \left\| \mathbf{p}_i^{(a)} - \mathbf{p}_i^{(b)} \right\|^2$, where *a, b* denote the a-th and b-th splines subject to comparison.

Affine-invariant description and rapid classification. In the sequel, two problems arise: (a) the comparison and classification of curves must be invariant to possible affine transformations and (b) we should indicate a way of rapid initial classification since it is impossible to compare a sample curve with all curves existing in the database. Affine-invariant comparison is addressed in literature using curve moments and Fourier descriptors. It can be seen that the former approach is computationally costly but is reported to be relatively approximate, whereas the latter reduces computational cost however seems not to be a generic description for 2D curves.

As we mentioned above, a set of *m* sample points were used to describe the contour of an object. For each sample $\mathbf{s}_k$, k=0,...,m-1, the sequence $\mathbf{b}_k = \mathbf{s}_{xk} + j\mathbf{s}_{yk}$ is obtained, where $\mathbf{s}_{xk}$, $\mathbf{s}_{yk}$ denote the x,y coordinates for $\mathbf{s}_k$. The discrete Fourier factors for this sequence are obtained by

$$F_i = \sum_{k=0}^{m-1} \mathbf{b}_k \cdot \exp\left(-\frac{j2p \cdot i \cdot k}{m} \right), \quad i = 0,1,\ldots,m-1$$

If $\mathbf{b}'_k$ is a sequence obtained from $\mathbf{b}_k$ by scaling, translation, rotation and shift, then the discrete Fourier factors are given

$$\mathbf{P} = \begin{bmatrix} N_0(t'_1) + N_{n+1}(t'_1) & N_1(t'_1) + N_{n+2}(t'_1) & N_2(t'_1) + N_{n+3}(t'_1) & N_3(t'_1) & \cdots & N_n(t'_1) \\ N_0(t'_2) + N_{n+1}(t'_2) & N_1(t'_2) + N_{n+2}(t'_2) & N_2(t'_2) + N_{n+3}(t'_2) & N_3(t'_2) & \ldots & N_n(t'_2) \\ \vdots & & \vdots & & & \vdots \\ N_0(t'_m) + N_{n+1}(t'_m) & N_1(t'_m) + N_{n+2}(t'_m) & N_2(t'_m) + N_{n+3}(t'_m) & N_3(t'_m) & \cdots & N_n(t'_m) \end{bmatrix}$$

by

$$F_i' = a \cdot F_i \cdot \exp\left(j \frac{J - 2p \cdot i \cdot k_0}{m} \right) + \mathbf{b}_0 \cdot d(0)$$

and the normalized Fourier descriptors $\mathbf{v}_i = |F_i'|/|F_1'|$, i=2,3,...,m-1, are invariant to translation, rotation and starting point.

As it will be seen in the sequel, the normalized Fourier descriptors are fed into a neural network (NN). In order to keep the inputs of the NN reasonably small, we choose to use only the knot-points instead of all the sample points. Thus $\mathbf{v}$ is an $l\times1$ vector.

Although the normalized Fourier descriptors possess the aforementioned desirable properties, they seem to be a poor description for the contour curve of an object. For this reason, in this work, these descriptors are used only as an 'initial description' for the curve. The input sample curve is classified to one or more classes w.r.t. to Fourier descriptors, and then a fine match is performed using all curves belonging to these classes. This fine match is accomplished using curve moments [7,8].

In this case, each spline is parametrized in terms of its arc lengths s as $\mathbf{R}(s)=[x(s), y(s)]$ which is a known function of its control points. The (p,q) order moments are weighted by kernels w_j, so that

$$m(p,q)^{(j)} = \int_{s=0}^{S} x^p(s) \cdot y^q(s) \cdot w_j(x,y)\, ds \,.$$

By appropriate choise of the kernels, it can be seen that the affine parameters $\mathbf{L}$, $\mathbf{c}$ aligning two curves, i.e. $\mathbf{r}(t')^{(a)} = \mathbf{L} \cdot \mathbf{r}(t')^{(b)} + \mathbf{c}$, can be estimated from their moments up to order two [7].

3. NEURAL NETWORK CLASSIFICATION

Along the lines of the previous section, it is possible for a given set of curve prototypes to determine which one matches best a given curve indepedently of affine transformations. At first, using groups of curve prototypes, we define primary object classes (e.g., airplanes, cars, vases etc.), which can be further organized in an object class database. Hence, the problem of classifying a sample curve to a specific class reduces into locating the best match between this sample curve and the set of all prototypes. Note, however, that although the B-spline representation is affine invariant, it is essential that each class contains several prototypes depicting different object instances or variations, different views or even views in different level of detail. Consequently, a very large amount of curve prototypes would be used in a practical system, making the procedure of direct comparison with all available prototypes extremely time consuming.

For this purpose, a neural network approach is used in order to constrain the search procedure into a small subset of object classes. In particular, the representation of curve prototypes (normalized Fourier descriptors) is used as an input to a feedforward NN, and a network output is assigned to each primary object class. The network attempts to implement a mapping between an input pattern $\mathbf{v}=[v_1,v_2,\ldots,v_N]^T$ and a desired output pattern $\mathbf{d}=[d_1,d_2,\ldots,d_C]^T$. A neural network with two hidden layers is used, as shown in Figure 1, with N input neurons, N_1 and N_2 neurons in the first and second hidden layer respectively, and C neurons in the output layer. Neurons of successive layers are interconnected through weights, so that for each neuron s, the net input is determined from $n_s = \sum_i a_i w_i$, where a_i is the output of the i-th neuron of the previous layer, w_i the weight connecting this neuron with neuron s, and the summation is evaluated over all neurons of the previous layer. The net input is then transformed by the sigmoid activation function

$$o_s = f(n_s) = \frac{1}{1+e^{-\ln_s}}$$

where o_s is the output of neuron s and $?$ is a gain parameter [9]. In the training stage, the B-spline representation $\mathbf{v}^{(p)}$, p=1,…,M of a set of M curve prototypes is fed as input to the NN, while the desired output $\mathbf{d}^{(p)}$, p=1,…,M is determined by setting the component of $\mathbf{d}^{(p)}$ that corresponds to the curve prototype class equal to one and all the other components to zero. The Levenberg-Marquardt method is used for training, attempting to minimize the sum-squared error

$$E = \sum_{p=1}^{M} \left\| \mathbf{d}^{(p)} - \mathbf{o}^{(p)} \right\|^2 = \sum_{p=1}^{M} \sum_{i=1}^{C} (d_i^{(p)} - o_i^{(p)})^2$$

between the desired and actual output patterns $\mathbf{d}_p$ and $\mathbf{o}_p$ respectively. The minimization is performed by updating the weights connecting neurons of successive layers and re-evaluating the outputs and the sum-squared error in an iterative way.

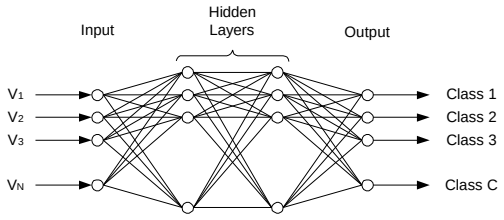


Figure 1. Neural network architecture used for object classification.

In the allocation stage, the B-spline representation $\mathbf{v}=[v_1,v_2,\ldots,v_N]^T$ of a test curve is used as input to the NN. Since one network output corresponds to each object class, representing the classification result of the input curve into the respective class, the input curve is typically classified to the object class that corresponds to the maximum network output. However, since all the other output components are not equal to zero (as is the case for the curves of the training set), misclassification might occur in some cases if the network outputs are close to each other. For this reason, R classes are selected for each input curve, corresponding to the network outputs with the maximum values, where R corresponds to a small percentage of the total number of classes, M. This set of classes is then used for the matching procedure, in order to select the class that provides the best match. Curve matching in this case is performed between the input curve and all instances and variations of the prototype curves of the R selected classes, resulting in a more robust and reliable classification. Alternatively, when the classes become very populated, curve matching is performed using a small number of representatives for each of the R classes.

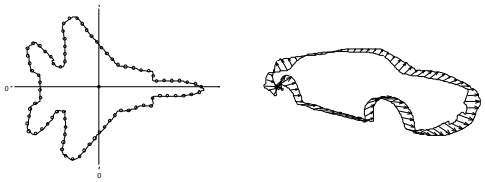
4. EXPERIMENTAL RESULTS

The aforementioned methodology for image classification and retrieval has been tested using an MPEG video database containing video sequences of total duration 4 hours. Each

sequence is partitioned into video shots and feature vectors are constructed for each frame, containing color and motion information. A small number of key frames and shots is then selected used the techniques described in [3] so that subsequent processing is constrained to a subset of the original sequences. Object contours are obtained through color and motion segmentation, as depicted in Figure 2. Reallocated knot-points are then derived for each curve so that the correspondence between starting points of different curves is preserved, as shown in Figure 3. The Fourier descriptors of the reallocated knot-points are used as input in the NN.

(a) (b) (c)

Figure 2. Extraction of object contours through segmentation: (a) initial image, (b) segmentation result, (c) object contour.



(a) (b)

Figure 3. (a) Reallocated knot-points, (b) Knot-point matching between two distinct car curves.

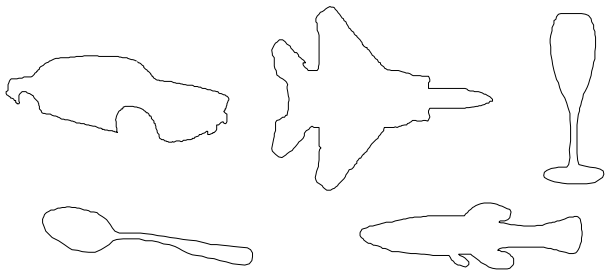


Figure 4. Sample prototype curves corresponding to distinct object classes and used for training.

(a) (b)

Figure 5. Sample input curves. (a) input curve not belonging to the training set, (b) transformed curve from the training set.

Object Class	NN Classification	Curve Matching
Cars	10/10	9/10
Airplanes	10/10	10/10
Glasses	9/10	8/9
Spoons	9/10	9/9
Fish	10/10	9/10
Total	**48/50**	**45/48**

Table 1. Classification results.

Five object classes are defined for the experiments in this paper, corresponding to cars, airplanes, glasses, spoons and fish. A sample prototype curve for each object class is illustrated in Figure 4, while 10 curves per class are actually used for training and 10 different curves per class are used for classification testing. One such sample curve, not belonging to the training set, is shown in Figure 5(a), while an affine-transformed prototype curve is shown in Figure 5(b). It should be mentioned that due to the invariance properties of the Fourier descriptors, curves subjected to any affine transformation give exactly the same classification result, hence classification using transformations of the curves belonging to the training set is 100% successful. After NN training with the training set consisting of 50 curves, classification is tested using the 50 curves of the test set. Two classes (R=2) are derived for each input curve, corresponding to the NN outputs with the maximum values. Curve matching using all 10 curve instances for each of the two classes is then employed in order to select the best matching class. In particular, 20 metric distances are calculated and the input curve is assigned the class corresponding to the minimum distance. The results are shown in Table 1.

5. CONCLUSIONS – FUTURE WORK

A system for content-based image retrieval from image/video databases based on object contours has been presented in this paper, using B-splines for affine invariant contour representation, and a neural network for supervised classification of objects into prototype object classes. This technique of locating an initial number of candidate object classes, and then refining the selection with curve matching results in a very fast and accurate implementation. Furthermore, higher level classes can be defined by combining primary classes, providing the ability to obtain a high level of abstraction in the representation of each video sequence. This prospect is currently under investigation.

REFERENCES

[1] D.Zhong and S.-F.Chang, "Video Object Model and Segmentation for Content Based Video Indexing," *Int. Conf. Circuits and Systems*, Hong Kong, June 97.

[2] N.Vasconcelos and A.Lippman, "Towards Semantically Meaningful Feature Spaces for the Caracterization of Video Content," *Proc. ICIP*, Santa Barbara, USA, Oct. 97.

[3] N.Doulamis, A.Doulamis, Y.Avrithis, and S.Kollias, "Video Content Representation Using Optimal Extraction of Frames and Scenes," accepted in *ICIP*, Chicago, USA, Oct. 98.

[4] Z.Lei, Y.Chan, and D.Lopresti, "Image Curvelet Feature Extraction and Matching," *Proc. ICIP*, Santa Barbara, USA, Oct. 97.

[5] M.Swanson and A.Tewfik, "Affine-Invariant Multiresolution Image Retrieval Using B-Splines," *Proc. ICIP*, Santa Barbara, USA, Oct. 97.

[6] F.S.Cohen, Z.Huang, and Z.Yang, "Invariant Matching and Identification of Curves using B-Splines Curve Representation," *IEEE Trans. Image Proc.*, Vol.4, No.1. Jan. 95.

[7] Z.Huang and F.S.Cohen, "Affine-invariant B-Spline Moments for Curve Matching," *IEEE Trans. Image Processing*, Vol.5, No.10. Oct. 96.

[8] K.Arbter, W.E.Snyder, H.Burkhardt and G.Hirzinger, "Application of affine-invariant fourier descriptors to recognition of 3D- objects," *IEEE Trans.Pattern Anal. Mach.Intell*, Vol.12, No.7, pp. 640-647, July 1990.

[9] D. R. Hush and B. G. Horne, "Progress in Supervised Neural Networks," *IEEE Signal Processing Magazine*, Jan 1993.