Feature weighting methods for abstract features applicable to motion based video indexing

Feature Weighting Methods for Abstract Features Applicable to Motion based Video Indexing

Ashfaqur Rahman, Manzur Murshed, and Laurence S. Dooley
Gippsland School of Comp. & IT, Monash University, Churchill Vic 3842, Australia
{Ashfaqur.Rahman, Manzur.Murshed, Laurence.Dooley}@infotech.monash.edu.au

Abstract

Content based labels, associated with image sequences in contemporary video indexing methods, can be textual, numerical as well as abstract, including colour-histograms and motion co-occurrence matrices. Abstract features or indices are not explicitly numeric entities but rather are composed of numeric entities. When multiple abstract features are involved, distance metrics between image sequences need to be weighted. Most feature weighting methods in the literature assume that the space is numeric (either discrete or continuous) and so not applicable to abstract feature weighting. This paper elaborates some feature weighting methods applicable to abstract features and both binary (feature selection) and real-valued weighting methods are discussed. The performance of different feature selection and weighting methods are provided and a comparative study based on motion classification experiments is presented.

1. Introduction

With the advances in communications and multimedia computing technology in the areas of computer vision, medical imaging, audio-visual archives, digital TV, and road traffic surveillance, there is a urgent need to have a visual management system or tools to assist users to efficiently index, store, and retrieve information from digital video databases. Video indexing is the process of attaching labels to sequences to enable efficient retrieval to be achieved, instead of the raw data. Manual annotation is one indexing solution; however it is generally restricted to small sized video databases. For large video collections, there is an obvious need for automated indexing and retrieval of video documents based upon their content.

Contemporary indexing methods [1]-[2] address the problem of automatic annotation using indices based on colour, texture, structure, and motion content of videos. The indices which are extracted can be textual, numeric (discrete or continuous) or *abstract features*. Abstract feature or index is not a numeric entity itself rather a composition of numeric entities. Examples include colour-histograms based on colour distribution over the video sequences and motion co-occurrence matrices for motion based video indexing. Given a video sequence, multiple abstract features are extracted from each sequence for indexing. Divergence between two sequences for each type of abstract feature is then mapped to a numerical value.

The conventional way to obtain a distance between a query video and database videos is to measure the Euclidian summation of feature divergences. The Euclidian distance assigns an equal weight to each feature, however not all features have equal distinguishing power among the many different types of video sequence, so a weight assignment, either binary or real valued is required to differentiate the features. The former is known as feature selection method and the later as feature weighting method.

The problem of applying feature selection or weighting methods directly for abstract features is that most of these methods assume that feature space is discrete or continuous which abstract features fail to meet. There are methods [4]-[9] that instead of directly working on feature values operate on distance or similarity measures. As abstract features have well defined numeric distances, these later methods are applicable for abstract feature weighting. This paper explores some of these methods and experimental results are presented by applying these methods in motion classification experiments.

The remainder of this paper is organized as follows. Section 2 provides a brief review of different feature selection and weighting methods applicable to abstract features. Section 3 discusses abstract features used in motion based video indexing, while Section 4 describes the experimental platform. Some experimental results with different feature selection and weighting methods applicable to motion based abstract features are discussed in Section 5, while Section 6 concludes this paper.

2. Feature selection and weighting methods

Algorithms for feature selection or feature weighting can be broadly classified into two main categories depending on whether the method employs feedback from

Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04)
0-7695-2108-8/04 $ 20.00 © 2004 IEEE

the subsequent performance of the learning algorithm. These two approaches are termed as wrapper (uses feedback) and filter (no feedback) methods. Although this categorization is defined in the context of feature selection, it is equally applicable to feature weighting problems.

2.1. Filter methods

Filter methods [7]-[9] do not use feedback, and so are pre-selection methods independent of the subsequent learning or classification algorithm to be applied. The data is first analysed, for instance using statistical techniques, to determine the relevance of features to describe the data. Weights (binary or real) are set according to the relevance and are then used in the training. Conversely, wrapper methods [4]-[6] are feedback methods that incorporate the Maximum Likelihood algorithm in the feature evaluation process. Although filter methods are faster, weight assignment in wrapper methods is to optimize final classification problem. In general, wrapper methods attempt to optimize directly the predictor performance so that they can provide better performance. It is in this context that this paper focuses on wrapper methods for feature selection and weighting.

2.2. Wrapper methods for feature selection

The wrapper methods applicable to abstract feature selection include forward selection (*FS*), backward elimination (*BE*), genetic algorithms (*GA*) and *BSMT* (Basic sort merge technique). Forward selection [5] starts with an empty set of features and successively adds individual features, usually following a variant of a greedy algorithm, terminating when no improvement is possible. At an intermediate step a new feature, the addition of which gives best classification performance, is added to the current selected list. Backward elimination [5], which does the reverse, starts with the full set of features and heuristically subtracts individual features until deletion of a feature provides no further improvement.

Genetic algorithm [4] performs a randomized search and it is not very susceptible to getting stuck in local minima. Moreover, the crossover operator has the effect of merging solutions whilst preserving the already successful feature selections. The *GA* population is coded as simple vectors of binary genes, where 1 represents relevant features. The fitness of solutions is evaluated by running classification experiments on the training data set using only the features corresponding to 1s in the chromosome, and returning the classification accuracy as the fitness. During the evaluation of the *GA*, training data set is used for calculating classification accuracy and the final set of selected features are applied on test set to evaluate performance.

BSMT (basic sort merge technique) [5] combines the methods of forward selection, backward elimination and genetic algorithms. To avoid irrecoverable adding or subtracting, it always operates on some representation of the original feature space, so that at each step every feature has an opportunity to impact upon the selection. To avoid heuristic randomness, at each step a greedy algorithm is used to govern subset formation. *BSMT* is divided into two parts: the creation of a tree of feature subsets and the manipulation of the tree to create a feature subset of desired cardinality or accuracy.

The tree creation part starts with all the features placed in single subsets. At every step, the current subsets are used separately to classify the training data set and the subsets are sorted according to their classification accuracy. These sorted subsets are then merged pair wise to form new subsets for the next iteration. The process continues as long the number of feature subsets is more than one. Once the tree is formed, the feature selection procedure starts in the opposite direction i.e. from the root. To select r features the leftmost branch with $2^{\lceil \log_2 r \rceil}$ features is selected first. From this branch $2^{\lceil \log_2 r \rceil} - r$ features are pruned using an iterative technique similar to backward elimination. A subset of features, elimination of which provides best classification accuracy, is pruned from the list at each step, until the total number of features in the subset equals r.

All the abovementioned wrapper methods are applied for abstract feature selection in the context of motion classification experiments and summarized in Section 5.

2.3. Wrapper methods for feature weighting

Feature weighting using wrapper methods fall into two categories. The first one maintains a set of real valued weights for all the features and updates those weights iteratively to optimize classification performance. The second category is the hierarchical feature weighting (HW) approach.

Genetic algorithms [4] are a suitable platform for the first category. The *GA* approach for feature weighting is similar to feature selection except that a chromosome is decoded in a different manner. In feature weighting, a group of b consecutive genes in a chromosome is decoded to calculate the feature weight. For a total of n features the chromosome length = nb. The process starts with a random population, then all chromosomes are first decoded to calculate feature weights and each chromosome's fitness is measured by the classification accuracy of the training set using corresponding feature weights. A mating pool is then filled using the best fit chromosomes for subsequent crossover and mutation operations. GAs actually perform a randomized search and so are less susceptible to becoming stuck in a local minima.

The second category of wrapper methods is hierarchical feature weighting (HW) [3]. In *HW* instead of weighting all the features at a time, weights for correlated features are first computed and correlated features are placed in groups. Among these groups correlated ones are weighed and placed in higher level groups and the process continues. It is assumed that correlation among features is well defined. In [3] the hierarchical weighting method is explained in detail in the context of motion classification experiments.

This paper summarizes the performance, based on empirical evaluation, of the abovementioned feature weighting wrapper methods applicable to abstract features. Motion classification experiments are used for performance evaluation of these methods.

3. Abstract motion features

In the field of computer vision, extensive research has been devoted to recognize different types of motion. Temporal textures represent a major motion component in many real world sequences. Temporal textures are sequences of images of moving scenes that exhibit some kind of temporal regularity. These include sea-waves, smoke, foliage, traffic scenes etc.

In [3],[10]-[12] the problem of temporal texture classification was addressed using motion co-occurrence matrices. Co-occurrence matrices encode the occurrence frequency of pair of motion measures, computed between successive frames, over a frame sequence. Multiple co-occurrence matrices [3] are computed to encode motion distribution of an image sequence. Not all these co-occurrence matrices contribute equally towards motion based classification. Accuracy varies on selection and weighting of these co-occurrence matrices. In this paper, performance of different feature (co-occurrence matrix) selection and weighting methods are evaluated by their motion classification accuracy.

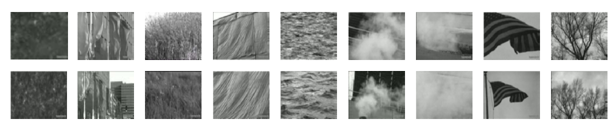
To calculate feature distance, a divergence measure is required between videos for each co-occurrence matrix type feature. Given two identical type of co-occurrence matrices associated with two different video clips v_1 and v_2 respectively, the Kullback-Leibler (*KL*) divergence as defined in [11]-[12] and utilized in this paper, approximates the amount of information lost when the co-occurrence statistics, associated with video clip v_1 is replaced by that of clip v_2.

Motion classification experiment performed in [3] on block based videos (MPEG-1/2/4 and H.26X) is used as the experimental platform. A total of (5×2) 10 temporal and (4×2) 8 spatial i.e. 18 co-occurrence matrices are computed for each video clip. The distance between two video clips for a particular type of co-occurrence matrix is mapped to a numerical value by KL divergence as stated above. Collectively co-occurrence matrices form a group of abstract features. Feature selection and weighting methods are implemented for selecting and weighting these co-occurrence matrices.

4. Experimental platform

4.1. Experimental inputs

A series of sequences containing a representative set of different temporal textures, including boiling water, waving flag, and wind swept grass, was incorporated into the video database. Nine different temporal textures were used as shown in Figure 1. For each texture type, two MPEG image sequences were partitioned into 14 video clips (the exception being type C, where 12 clips were used) comprising twenty 352×240 pixel frames using 16×16 pixel macroblocks. Out of each temporal texture type 7 video clips (For class C, 6 video clips) were chosen to build up a representative training set of image sequences. The remaining image sequences build up the test set. Only *I* and *P* frames were considered and the motion vectors were computed using full search process with a maximum displacement of $\delta = \pm 7$ pixels.



A. Boil B. Flag C. Grass D. Plastic E. River F. Smoke G. Steam H. Stripe I. Trees

Figure 1. Two representative video clips for the nine different temporal textures used in the experiments.

4.2. Classification accuracy

Once the database was populated with video clips and selection or weighting of co-occurrence matrices was done, each video clip was used as a query example to retrieve the *k*-nearest similar clips. As queries could be supplied from the database itself, the same video as a reply was ignored for performance evaluation. The *satisfaction* level of any query reply was measured as

$$\begin{cases} 1 - j\dfrac{\beta}{k}, & j < k; \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where *j* is the number of misclassified replies and $\beta < 1$ is the penalty reduction factor. The satisfaction level (2) signifies no bias in the order of reply, but does introduce a decreasing level of penalties, provided one reply is correctly classified. In all the experiments, the parameters $k = 3$ and $\beta = 0.25$ were used. Let $s^{\xi}(j)$ be the aggregated satisfaction level of all video clips of temporal texture type

Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04)
0-7695-2108-8/04 $ 20.00 © 2004 IEEE

j using feature selection or weighting method ξ. Thus $\chi_\xi = \bar{s}^\xi - \sigma(s^\xi)$ represents classification accuracy for ξ.

5. Experimental results

5.1. Feature identity

Feature selection and weighting methods are applied on a set of 18 co-occurrence matrices for motion classification experiments. Among them 5 temporal and 4 spatial magnitude co-occurrence matrices, 5 temporal and 4 spatial angular co-occurrence matrices correspond to different neighbouring position. Table 1 provides a mapping of these features to corresponding neighbouring positions.

Table 1. Co-occurrence matrix identity

Temporal magnitude/angular co-occurrence matrices			Spatial magnitude/angular co-occurrence matrices		
1/6	2/7	3/8	11/15	12/16	13/17
4/9	5/10	N/A	14/18	N/A	N/A
N/A	N/A	N/A	N/A	N/A	N/A

Table 2. Selected features for different feature selection methods

Feature selection method	Features selected
Forward selection	11,13,14
Backward elimination	10,12,13,18
GA	11,12,13
BSMT	11,13,14

5.2. Feature selection methods

The performance of feature selection methods are presented in Table 2 and Table 3. It can be verified from Table 2 that all the selected features belong to the spatial domain which implies the significance of spatial co-occurrence matrices over temporal co-occurrence matrices. From the performance point of view *BSMT* performs the best on the basis of test case performance. Selected features in *BSMT* are the same as *FS*. This can be attributed to the fact that *BSMT* selects the best among *GA*, *FS* and *BE*. Performance of all feature selection methods is better than when all features are used.

Table 3. Performance of different feature selection methods

Feature selection method	Training case	Test case
All features	68.40	76.97
FS	81.77	84.96
BE	80.85	78.66
GA	81.40	81.05
BSMT	81.77	84.96

5.3. Feature weighting methods

Feature weights obtained by *GA* and *HW* are presented in Table 4 and Table 5 respectively. The performances of the feature weighting methods are presented in Table 6. For hierarchical weighting method, the weights are first calculated among magnitude and angular co-occurrence matrices for each spatial and temporal neighbour. At a second level hierarchy weights are calculated among temporal neighbours as well as among spatial neighbours using a *GA* method. Finally weight is calculated for temporal dimension and spatial dimension.

It can be observed from both weighting methods that spatial features are getting more weights than temporal features which agree with the results of feature selection methods where spatial features were selected. For hierarchical weighting magnitude co-occurrence matrices are getting higher weights mostly compared to angular co-occurrence matrices. From performance point of view both methods perform almost similar. For *HW* additional weights are required to be calculated compared to *GA*. On the other hand time complexity of *HW* is less than *GA* because search space is smaller in *HW* at each hierarchy. Performance of all feature weighting methods is better than when all features are given equal weights.

Table 4. Feature weights for GA

Temporal magnitude		
0.0067	0.0201	0.0201
0.0067	0	0
0	0	0

Spatial magnitude		
0.1611	0.1812	0.2013
0.1342	0	0
0	0	0

Temporal angle		
0	0	0
0	0.0268	0
0	0	0

Spatial angle		
0.0872	0.0537	0.0671
0.0336	0	0
0	0	0

Table 5. Feature weights for HW. Angular weights can be obtained by subtracting corresponding magnitude weight from 1.

Temporal magnitude				Spatial magnitude		
0.5188	0.5625	0.8813		0.8438	0.8188	0.9063
0.3063	0.6	0	Step 1	0.6188	0	0
0	0	0		0	0	0

Temporal neighbours				Spatial neighbours		
0.0588	0.2941	0.2157		0.0714	0.4	0.2429
0.1569	0.2745	0	Step 2	0.2857	0	0
0	0	0		0	0	0

Temporal		Spatial
0.0438	Step 3	0.9562

Table 6. Performance of different feature weighting methods

Feature weighting method	Training case performance	Test case performance
All features used	68.40	76.97
GA	82.78	79.66
Hierarchical weighting method	82.75	82.65

6. Conclusions

This paper has provided a broad series of guidelines for applying feature selection and weighting methods to abstract features. Performance of different feature selection and weighting methods and a comparative study has been provided based on motion classification experiments. Although experimental results are obtained based on motion co-occurrence matrices, they are applicable to other abstract features including colour histograms. Experimental results clearly show that selection of relevant features or feature weighting provide much superior results than when all features are considered.

7. References

[1] C. G. M. Snoek and M. Worring, "A review on multimodal video indexing," Proceedings of ICME 2002, vol. 2, pp. 21-24., Aug 2002.

[2] M.R. Naphade and T.S. Huang, "Extracting semantics from audio-visual content: the final frontier in multimedia retrieval," IEEE Transactions on Neural Networks, vol. 13, pp. 793 -810, July 2002.

[3] A. Rahman, M. Murshed and L. S. Dooley, "A new video indexing and retrieval method for temporal textures using block-based co-occurrence statistics," Tech. Report TR-2003/5, Gippsland School of Computing and IT, Monash University, 2003.

[4] J. Jarmulak and S. Craw,"Genetic Algorithms for Feature Selection and Weighting," In Proceedings of the IJCAI'99 workshop on Automating the Construction of Case Based Reasoners, pp. 28-33, 1999

[5] Y.Liu, and J. R. Kender, "Sort-Merge Feature Selection for Video Data." SDM 2003.

[6] D. Wettschereck, D. Aha, and T. Mohri, "A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms," Artificial Intelligence Review, pp. 273--314, 1997.

[7] H. K. Lee, and S. I. Yoo, "A neural network-based image retrieval using nonlinear combination of heterogeneous features," Proceedings of the 2000 Congress on Evolutionary Computation, vol. 1, pp. 667-674, July, 2000.

[8] E. C. C. Tsang, S. C. K. Shiu, and X. Z. Wang, M. Lam," Clustering and classification of cases using learned global feature weights," IFSA World Congress and 20th NAFIPS International Conference, vol. 5, pp. 2971-2976, 2001.

[9] D. S. Yeung, and X. Z. Wang, "Improving performance of similarity-based clustering by feature weight learning," IEEE transactions on pattern analysis and machine intelligence, vol. 24, pp. 556-561, April, 2002

[10]P. Bouthemy and R. Fablet, "Motion characterization from temporal cooccurences of local motion-based measures for video indexing," Int. Conf. on Pattern Recognition (ICPR'98), vol. 1, pp. 905-908 , 1998.

[11]R. Fablet and P. Bouthemy, "Non parametric motion recognition using temporal multiscale Gibbs models," Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, vol. 1, pp. 501-508, 2001.

[12]R. Fablet, P. Bouthemy, and P. Perez, "Nonparametric motion characterization using casual probabilistic models for video indexing and retrieval," IEEE Trans. on Image Processing, vol. 11, pp. 393 -407, 2002.