MULTI-MODAL CLASSIFIER FUSION FOR VIDEO SHOT CONTENT RETRIEVAL

Fabrice Souvannavong, Bernard Merialdo and Benoit Huet

Département Communications Multimédias
Institut Eurécom
2229, route des crêtes
06904 Sophia-Antipolis - France
(Fabrice.Souvannavong, Bernard.Merialdo, Benoit.Huet)@eurecom.fr

ABSTRACT

In this paper we present a new chromosome to solve the problem of classifier fusion using genetic algorithm. Experiments are conducted in the context of TRECVID. In particular we focus on the feature extraction task that consists in retrieving video shots expressing one of predefined semantic concepts. Three modalities (visual, textual and motion) and two features per modality are used to describe the content of a video shot. Thus, we require fusion techniques to efficiently manage all these heterogeneous sources of information. A first step achieves the classification per feature and concept, then a genetic algorithm is used to efficiently fuse the output of all classifiers. For this purpose, a dynamic binary tree is proposed to model the novel chromosome for hierarchical fusion.

Keywords: *genetic algorithm, classification, fusion, binary tree, video shot analysis, feature extraction, content retrieval*

1. INTRODUCTION

With the growth of numeric storage facilities, many documents are now archived in huge databases or extensively shared on the Internet. The advantage of such mass storage is undeniable, however the challenging tasks of automatic content indexing, retrieval and analysis remain unsolved, especially for video sequences. TRECVID [1] stimulates the research in this area by providing common datasets for evaluation and comparison of new techniques and systems. One specific task is the feature extraction that consists in retrieving shots containing a given semantic concept. This challenging task is important for future information retrieval systems which are more likely to work at the semantic level of the content than just its visual aspect.

The semantic content is carried by many modalities and many features per modalities. And all this information must be fused to make final decisions, either for classification or retrieval. The fusion can be integrated at various places during the retrieval process. Usually it happens at the beginning (feature fusion) or at the end (classifier fusion). The objective of feature fusion is to combine multiple features at an early stage to construct a single model. In an ideal situation, feature fusion should work for all concepts, since there is no loss of information. However, practical considerations, such as limited number of training examples, limited computational resources, and the risk of over-fitting the data, require an alternate strategy. Classifier fusion is an active research field [9, 8, 15, 13] and we propose an approach using genetic algorithms. In particular we introduce a new chromosome based on binary trees to represent the fusion mechanism. Contrary to existing approaches using genetic algorithms for fusion [9], the structure of the tree is dynamic to select optimal operators and operands.

Experiments are conducted in the context of TRECVID feature extraction task. This task consists in retrieving shots expressing a specific semantic concept in a video database. The fusion is then used to compute a unique detection score from the output of many classifiers.

The first section presents the motivations in using genetic algorithms for fusion. Then, we present features extracted from video shots. The next section presents the classification process to compute first-level detection scores. Next, the fusion algorithm to compute final detection scores is presented. Finally experimental results are discussed and we conclude with future works.

2. MOTIVATIONS

Complex semantic concepts that need to be retrieved require the analysis of many features per modalities. However it is far from trivial to fuse all this information. The fusion mechanism can be activated at the different stages of the classification. Generally, the fusion is applied on signatures (feature fusion) or on classifier outputs (classifier fusion). Unfortunately, complex signatures obtained from fusion on signatures are difficult to analyze and it results in classifiers that are not well trained despite of the recent advances in machine learning based on the concept of Support Vectors [7]. Therefore, the fusion of classifier outputs remains an important step of the classification task.

Using simple fusion operators like sum, product, min and max, can provide good performances and we propose a novel hierarchical fusion mechanism using these simple operators to improve performances. The hierarchy is represented by a binary tree which allows to model all possible formulas with a fixed number of inputs and operators on two operands. Unfortunately, it is difficult to carry out exhaustive experiments to select the structure, the appropriate operators and weights. For this purpose, we propose to use genetic algorithms.

3. SHOT FEATURES

We distinguish three types of modalities: visual, text and motion features.

3.1. Visual features

To describe the visual content of a shot, we extract features on its key frame. Two visual features are selected for this purpose: Hue-Saturation-Value color histograms and energies of Gabor's filters [10]. In order to capture the local information in a way that reflects the human perception of the content [2, 5], visual features are extracted on regions of segmented key-frames [3]. Then, to have reasonable computation complexity and storage requirements, region features are quantized and key-frames are represented by a count vector of quantization vectors. At this stage, we introduce latent semantic indexing to obtain an efficient region based signature of shots. Finally, we combine the signature of the key-frame with the signatures of two extra frames in the shot, as it is described in [14], to get a more robust signature.

3.2. Text features

The text or voice are important features. Both help to bridge the gap from low-level features to the semantic content by providing a direct information about the semantic content. Text features are based on the automatic speech recognition text provided by LIMSI [4] for TRECVID data sets.

First of all, words are stemmed with the widely used Porter's algorithm [12]. Then a dictionary of 2,000 words is created and shots are described by a count vector of the dictionary entries. However, a shot is not a semantic unit, then few words occur in a shot and relevant words might be in surrounding shots. To deal with this synchronization problem, basic text signatures of surrounding shots are included into the current shot signature. This is equivalent to compute a signature over a scene defined as the set of shots that surround the current shot.

3.3. Motion features

For some concepts like *basket scored, people walking/running, violence* or *airplane takeoff*, it is useful to have an information about the activity present in the shot. Two features are selected for this purpose: the camera motion and the motion histogram of the shot. For sake of fastness, these features are extracted from MPEG motion vectors. The algorithm presented in [16] is used to estimate the camera motion of a frame. The camera motion is approximated by a six parameter affine model. We then compute the average camera motion over the shot. The estimated camera motion is subtracted from macro-block motion vectors to compute the 64 bins motion histogram of moving objects in a frame. Then, the average histogram is computed over frames of the shot.

4. CLASSIFIERS

We focus our attention on general models to detect TRECVID features. We have decided to compute a detection score per low-level feature at a first level. The genetic algorithm presented in the next section will then take care of the fusion of all detection scores at a second level.

The first level of classification is achieved with either the k-nearest neighbor classifier or the support vector machine classifier. In the particular case of text features, we also propose to compute a detection score based on a set of keywords per concept.

4.1. K-nearest neighbors

Since we have no information about the distribution shape of the data, we find natural to use the K-NN classifier as a baseline. Given a shot i, its N nearest neighbors in the training set are identified $(trshot_k), k = 1..N$. Then it inherits from its neighbors a detection score as follows:

$$D_f(shot_i) = \sum_{k=1}^{k=N} cosine(shot_i, trshot_k) * D_f(trshot_k)$$

Where detection scores of training shots, $trshot_k$, are either 1 if the concept f is present or -1 if not.

In order to optimize classifier performances, the algorithm finds the most appropriate number of neighbors for each couple formed by a low-level and a semantic feature. In the particular case of visual features, it also seeks for the best number of factors to be kept by the latent semantic indexing method [14].

K-NN classifiers were trained for all available low-level features: visual, text and motion features.

4.2. Support vector machine

Support vector machine classifiers compute an optimized hyperplane to separate two classes in a high dimensional space. We use the implementation SVMLight detailed in [6]. The selected kernel, denoted $K(.,.)$ is a radial basis function which normalization parameter σ is chosen depending on the performances obtained on a validation set. Let $\{sv_i\}, i = 1,...,l$ be the support vectors and $\{\alpha_i\}, i = 1,...,l$ corresponding weights. Then,

$$D_s(shot_i) = \sum_{k=1}^{k=l} \alpha_k K(shot_i, sv_k)$$

SVM classifiers are only trained on visual features.

4.3. Keywords detection

Using full text features as described in section 3, do not provide good classification performances with a k-NN classifier. The idea to efficiently use the text is then to identify important keywords for each concept and then compute a detection score based on the list of important keywords.

First of all, most occurring stemmed words are extracted for each concept from training data. We manually select words that are really related to the concept. Then, we estimate the probability that words related to a concept appear in surrounding shots. This a priori probability is further used to compute the final score. Let $P_f(shot_i + t)$ the probability to detect the concept f in the shot at $(i+t)$. Let $d_f(shot_i)$ the number of times words associated to the concept f occurs in the shot. Then

$$D_f(shot_i) = \sum_{t=-N}^{t=N} P_f(shot_i + t) \times d_f(shot_i + t)$$

5. FUSION

In this paper we propose to use a genetic algorithm to find the best combination of classifier outputs to compute the optimized output S_f. The next part defines our chromosome and remaining parts present the three involved steps in genetic algorithms: the initialization of the initial population of chromosomes, the genetic

transformations of chromosomes and the selection of best chromosomes.

5.1. Chromosome

Let $\{O_i\}, i = 1..N$ the outputs from N classifiers. The chromosome decomposes the fusion in three tasks. The first task of a chromosome is to normalize these outputs. We decide to work on probabilities and outputs are mapped into $[0..1]$. Four functions $\{f_n^j(.)\}, j = 1..4$ are proposed for this purpose: *shift, Gaussian v2, Gaussian v3, min-max*. The *shift* translates probabilities such that the minimum value is 0 and sets values higher than 1 to 1. The *Gaussian* operators model the distribution by Gaussian of mean 0.5 and variance $\frac{1}{4}$ or $\frac{1}{6}$. The *min-max* translates and scales values such that probabilities are in [0..1]. The second task consists in weighting probabilities with a priori probabilities a_i. The fusion is finally achieved in the third task with the help of four operators applied hierarchically on two operands. Selected operators are the mean, product, minimum and maximum. The hierarchy is a complete binary tree whose internal nodes are operators and whose leaves are input probabilities as represented in figure 1. Selected operators, except the mean, are associative, thus the binary tree also allows to represent operations on more than two operands. The structure of the tree is dynamic. It means that from one chromosome to another the structure might be different.

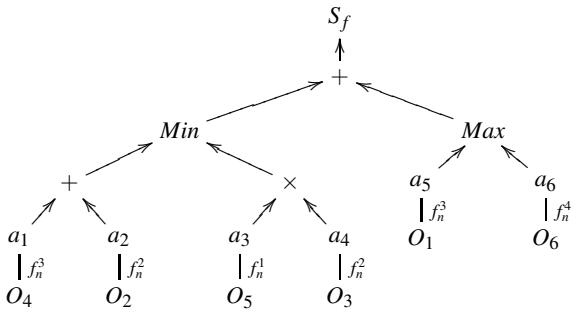

Fig. 1. Example of a hierarchy involved in a chromosome.

5.2. Initialization

The first step of genetic algorithm is the generation of the initial population. For this purpose, we need to uniformally pick-up chromosomes in the possible configuration space. While it is simple to uniformally pick-up random numbers to select a priori probabilities and operators, the generation of a random tree is not trivial. Most algorithms for generating random binary trees are based on string representations of binary trees. These algorithms, then, operates on specific grammars to generate valid strings [11]. Remy's algorithm is an exception to this general approach and we use it to create the population of chromosomes. To generate a complete binary tree with n internal nodes and $n+1$ leaves, the algorithm proceeds iteratively as follows:

1. suppose we have a binary tree with k internal nodes and $k+1$ leaves,
2. select a random node ($\diamond$) from the $2k+1$ nodes of the tree,
3. replace ($\diamond$) by a new node ($\star$) and randomly choose ($\diamond$) to be the left or right child of the new node. The other child is then a new leaf ($\circ$) (figure 2),
4. repeat the process until the $n+1$ leaves are in the tree.

5.3. Mutation and Crossover

Important functions of genetic algorithm are the mutation and crossover. They should allow to browse the entire space of possible configurations while attempting to generate chromosomes that fit to the problem. In our problem, the main question is how to modify the binary tree. We will process in a similar way as for its creation using Remy's identity (figure 2).

For the mutation: we randomly select a number of leaves to remove n_r. Next, we randomly remove n_r leaves from the tree and add n_r new random leaves to the tree. To add a new leaf we proceed as explained in the section 5.2 while to remove a random leaf we proceed as follows:

1. select a random leaf ($\circ$),
2. remove the leaf,
3. replace its parent ($\star$) by its brother ($\diamond$).

Each time a new leaf is added to the tree, a random operator is selected.

For the crossover between a father and a mother: we select a random node from the mother and count the number of leaves below the selected node, i.e. the selected subtree. As many leaves are removed from the father's tree to which is next added the mother's subtree.

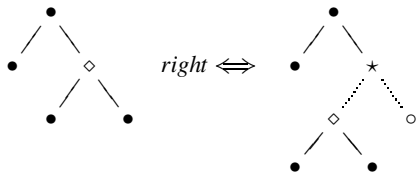

Fig. 2. Remy's identity to add or remove a leaf.

5.4. Fitness

The fitness function used in genetic algorithms allows to evaluate the fitness of a chromosome to the problem. The fitness values of chromosomes allow to sort the population and to select the ones on which will be applied mutation and crossover functions. In the framework of TRECVID, we are interested in information retrieval performances and in particular the mean precision measure. Thus, best chromosomes optimize the fusion with respect to the mean precision on a training set.

6. EXPERIMENTS

Experiments are conducted on the TRECVID 2003 and 2004 databases. It represents a total of over 120 hours of news videos. About 60 hours are used to train the feature extraction system and the remaining for the evaluation purpose. The training set is divided

Bill Clinton

modalities	input concepts	GA	SVM
visu	clinton	0.010	0.010
text	clinton	0.128	0.124
visu + text	clinton	0.152	0.124
visu + text	clinton + albrigth	0.155	0.124

Basket Scored

modalities	input concepts	GA	SVM
visu	basket	0.379	0.359
text	basket	0.075	0.067
visu + text	basket	0.474	0.225
visu + txt + motion	basket	0.411	0.226

Table 1. Evaluation of fusion performances. Comparison of mean precision values of two fusion systems and different inputs.

into two subsets in order to train classifiers and next the fusion parameters. The evaluation is realized in the context of TRECVID and we use the common evaluation measure from the information retrieval community: the mean precision.

The feature extraction task consists in retrieving shots expressing one of the following semantic concepts: *boat or ship, Madeleine Albright, Bill Clinton, train, beach, basket scored, airplane take-off, people running or walking, physical violence and road*. However the ground-truth provided with the training set is much more complete and is composed of 133 labels. We retained 19 labels related to the concepts to be retrieved and we expect the proposed genetic algorithm to correctly carry out the fusion. One classifier per raw feature (color, texture, text, motion, ...) and per label is trained. Depending on the feature to be retrieved, appropriate classifier outputs are provided to the fusion system to determinate the detection score of training shots.

Experiments show the benefit of including different modalities to the computation of a detection score (see table 1 for two concepts). We compare performances obtained over each modality, i.e. visual, text and motion, and performances obtained when modalities are combined. To complete the study, a comparison with a SVM classifier for fusion is provided. Results show the benefits of the genetic algorithm with the proposed dynamic tree structure. It efficiently fuse the information provided by the different modalities. However, the actual structure does not allow to discard incoherent inputs. This explains why including motion detectors to retrieve a *basket scored* reduces performances. Presented results encourage to add more inputs to the fusion, including face or object detectors, different color and texture models and sound analysis modules.

7. CONCLUSION

We presented a new chromosome using a dynamic binary tree to model the fusion function. A genetic algorithm was used to find best the chromosome that satisfies the task of video shot retrieval. Experimental results presented in the context of TRECVID feature extraction, show that the fusion algorithm was efficiently selecting appropriate normalization and fusion operators, weights and binary tree structures to optimize retrieval performances.

Future works will concern the structure of the tree and the family of fusion and normalization operators. The properties of tree nodes will be modified to locally take into account a priori probabilities and to discard some inputs.

8. REFERENCES

[1] TRECVID: Digital video retrieval at NIST. http://www-nlpir.nist.gov/projects/trecvid/.

[2] Chad Carson, Megan Thomas, and Serge Belongie. Blobworld: A system for region-based image indexing and retrieval. In *Third international conference on visual information systems*, 1999.

[3] P. Felzenszwalb and D. Huttenlocher. Efficiently computing a good segmentation. In *Proceedings of IEEE CVPR*, pages 98–104, 1998.

[4] J.L. Gauvain, L. Lamel, and G. Adda. The LIMSI broadcast news transcription system. *Speech Communication*, 37(1-2):89–108, 2002.

[5] Feng Jing, Mingling Li, Hong-Jiang Zhang, and Bo Zhang. An effective region-based image retrieval framework. In *Proceedings of ACM MM*, 2002.

[6] T. Joachims. *Advances in Kernel Methods - Support Vector Learning*, chapter 11 (Making large-Scale SVM Learning Practical). MIT Press, 1999.

[7] Josef Kittler. A framework for classifier fusion: Is it still needed? In *Proceedings of the Joint IAPR International Workshops on Advances in Pattern Recognition*, pages 45–56. Springer-Verlag, 2000.

[8] Ludmila I. Kuncheva. A theoretical study on six classifier fusion strategies. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 24(2):281–286, february 2002.

[9] Ludmila I. Kuncheva and Lakhmi C. Jain. Designing classifier fusion systems by genetic algorithms. *IEEE Transactions On Evolutionary Computation*, 4(4):327–336, september 2000.

[10] Wei-Ying Ma and Hong Jiang Zhang. Benchmarking of image features for content-based image retrieval. In *Thirty-second Asilomar Conference on Signals, System and Computers*, volume 1, pages 253–257, 1998.

[11] Erkki Mäkinen. Generating random binary trees: a survey. *Inf. Sci. Inf. Comput. Sci.*, 115(1-4):123–136, 1999.

[12] Martin F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[13] X. Shi and R. Manduchi. A study on bayes feature fusion for image classification. In *Proceedings of IEEE CVPR*, volume 8, june 2003.

[14] Fabrice Souvannavong, Bernard Merialdo, and Benoit Huet. Latent semantic analysis for an effective region-based video shot retrieval system. In *Proceedings of ACM MIR*, 2004.

[15] Belle L. Tseng, Ching-Yung Lin, Milind Naphade, Apostol Natsev, and John R. Smith. Normalized classifier fusion for semantic visual concept detection. In *Proceedings of IEEE ICIP*, 2003.

[16] Roy Wang and Thomas Huang. Fast camera motion analysis from MPEG domain. In *Proceedings of IEEE ICIP*, pages 691–694, 1999.