HIERARCHICAL GENETIC FUSION OF POSSIBILITIES

Fabrice Souvannavong and Benoit Huet

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

ABSTRACT

Classification and fusion are major tasks in many applications and in particular for automatic semantic-based video content indexing and retrieval. In this paper, we focus on the challenging task of classifier output fusion. It is a necessary step to efficiently estimate the semantic content of video shots from multiple cues. We propose to fuse the numeric information provided by multiple classifiers in the framework of possibility logic. In this framework, many operators with different properties were suggested to achieve the fusion. We present a binary tree structure to model the fusion mechanism of available cues and the genetic algorithms that are used to determine the most appropriate operators and fusion tree structure. Experiments are conducted in the framework of TRECVID feature extraction task that consists in ordering shots with respect to their relevance to a given class. Finally, we will show the efficiency of our approach.

Keywords: *fusion of classifier outputs, possibility logic, genetic algorithm, binary tree, semantic-based video content indexing*

1. INTRODUCTION

Multimedia digital documents are readily available, either through the Internet, private archives or digital video broadcast. Tools are required to efficiently index this huge amount of information and to allow effective retrieval operations. Unfortunately, most existing systems rely on the automatic description of the visual content through color, texture and shape features whereas users are more interested in the semantic content. To answer this need, the MPEG-7 [1] standard offers the possibility to describe the video content and in particular the semantic content. However, in practice an important gap remains between the visual descriptors and the semantic content. The main solution for now is the use of manually annotated content. Unfortunately, the

annotation task is very time consuming and error prone. New tools for automatic semantic video content indexing are highly awaited and an important effort is now conducted by the research community to automatically bridge the existing gap [12, 19].

Video content carries a huge quantity of visual information that we are still no able to fully and automatically identify. However, it is commonly assumed that the interpretation of the video content requires many features. Somehow these features have to be fused. However, it is far from trivial to fuse all this information. The fusion mechanism can be activated at 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 [9]. Therefore, the fusion of classifier outputs remains an important step of the classification task.

This work focuses on the fusion of classifier outputs for semantic-based video content indexing. We propose to assimilate classifier outputs to possibility values associated to an event, i.e. presence of a class in the shot. In the framework of possibility logic, the fusion of two sets of events is defined with respect to several fusion operators that have different properties. In this paper, we propose to model the fusion function by a binary tree whose internal nodes are fusion operators and leaves are associated to classifier outputs, i.e. events. Genetic algorithms are naturally introduced to find most appropriate fusion operators and tree structures. This work extends two aspects of our previous work presented in [18]. First, the fusion is conducted in the mathematical framework of possibility theory which includes more fusion operators. Secondly, trees have variable sizes which allows to fully discard some classifier outputs.

The paper is organized as follows. The first section presents the classification process of video shots into semantic classes. We detail the kind of visual features used

Fig. 1. General framework of the application.

to model video shot content and the classification system used to classify video shots into semantic classes. In the second section our fusion method named hierarchical genetic fusion of possibilities is introduced. The third section is dedicated to experiments in the framework of TRECVID 2005. Finally, we summarize the presented work and outline future work.

2. FIRST LEVEL CLASSIFICATION

This section describes the workflow of the semantic feature extraction process that aims to detect the presence in video shots of semantic classes, like building, car, U.S. flag, water, map, etc ... The section starts with a presentation of visual cues that will be used to compare and classify video shots. Then, image latent semantic analysis (ILSA) is applied on these features to obtain an efficient and compact representation of video shot content. Finally, support vector machines (SVM) are used to obtain the first level classification which output will then be used by the fusion mechanism. The overall chain is depicted in figure 1.

2.1. Visual cues

It is far from trivial to identify the right features to extract for a general purpose application such as video content indexing. Many features were proposed in the literature during the last two decades. In some of our recent work on video content indexing [16], we proposed to use a region-based approach with color and texture information. At the end, an image vector space model (IVSM) is obtained to efficiently represent video shot content.

First key-frames of video shots, that are provided by TRECVID, are segmented thanks to the algorithm described

in [5]. The algorithm is fast and provides visually acceptable segmentation. Its low computational requirement is an important criterion when we need to process a huge amount of data like the TRECVID database. Secondly, normalized HSV color histograms as well as mean and variance of 24 Gabor's filter response energies are computed for each region. Thirdly, the obtained vectors over the complete database are clustered to find the N most representative elements. The clustering algorithm used in our experiments is the well-known k-means. Representative elements are then used as visual keywords to describe video shot content. To do so, computed features on a single video shot are matched to their closest visual keyword with respect to the Euclidean distance or an other distance measure. Then, the occurrence vector of the visual keywords in the shot is build and this vector is called the raw signature of the shot.

The same process is applied on regions around salient points. They are detected thanks to the Haar wavelet transform as presented in [14]. The idea is to track and keep salient pixels at different scales. We then propose to build two rectangular regions around each salient point, one region on the left and the other on the right for vertical edges and one on the top and the other on the bottom for horizontal edges. The depth of rectangles is proportional to the scale level at which corresponding points were detected. We propose to have smaller rectangles for high frequencies. An illustration of both segmentation approaches is provided on figure 2.

2.2. ILSA

Image latent semantic indexing is an adaptation of a method used for text document indexing. It was originally introduced in [3] and it has now demonstrated its efficiency. In [17], we efficiently adapted LSA to image content. Start-

Fig. 2. Example of segmentation outputs.

ing from an IVSM, one can build the occurrence matrix of visual keywords in some training shots. The singular value decomposition of this matrix provides a new representation of video shot content where latent relationships can be emphasized.

Let C denote the occurrence matrix previously defined.

$$C = UDV^T \quad \text{where} \quad U^T U = V^T V = I \quad (1)$$

With some simple linear algebra we can show that a shot (modeled by a raw signature q) is indexed by p such that:

$$p = U^T q \quad (2)$$

p is called an ILSA signature. U^T is then the transformation matrix to the latent space. The SVD allows to discover the latent relationships by keeping only the L highest singular values of the matrix D and the corresponding left and right singular vectors of U and V . Thus,

$$\hat{C} = U_L D_L C_L^T \quad \text{and} \quad p = U_L^T q \quad (3)$$

The number of singular values kept drives the ILSA performance. On one hand, if too many factors are kept, the noise will remain and the detection of synonyms and the polysemy of visual terms will fail. On the other hand, if too few factors are kept, important information will be lost, resulting in performance degradation. Unfortunately, no solution has yet been found and only numerous experiments allow to find the appropriate factor number.

Video shots have now their visual content described by ILSA signatures on color, texture and two regions types. As far as the experiments reported in this paper are concerned, four signatures are used. The objective is then to deduce the semantic content of shots. Classification methods are appropriate tools for this task. They consist on automatically assigning labels to a given input vector. For this purpose a model is firstly created with respect to a training set. We propose to use support vector machines to solve our classification problem.

2.3. SVM

Support vector machines were widely used in the past ten years and they have been proven efficient in many classification applications. They have the property to allow a non

linear separation of classes with very good generalization capacities. They were first introduced by Vapnik [21] for the text recognition task. The main idea is similar to the concept of a neuron: separate classes with a hyperplane. However, samples are indirectly mapped into a high dimensional space thanks to a kernel function that respects the Mercer's condition [2]. This allows one to lead the classification in a new space where samples are assumed to be linearly separable. To this end, we use the implementation SVMlight detailed in [8]. The selected kernel, denoted $K(.,.)$ is a radial basis function which normalization parameter σ is chosen depending on the performance obtained on a validation set. Let $\{sv_i\}, i = 1, \dots, l$ be the support vectors and $\{\alpha_i\}, i = 1, \dots, l$ corresponding weights. Then,

$$\mathcal{H}_{SVM}(x) = \sum_{k=1}^{k=l} \alpha_k K(x, sv_k)$$

We take advantage of the validation procedure to find both SVM parameters and the best number of factors to be kept by the ILSA.

3. HIERARCHICAL GENETIC FUSION OF POSSIBILITIES

The classification of many features is still a challenging problem despite the recent advances in machine learning based on the concept of support vectors [9]. One solution is to fuse the information coming from elementary classifiers. We define an elementary classifier as a model build for a single feature.

Classifier fusion is an active research field [11, 10, 20, 15, 13]. First approaches select the best model among the set of inputs [13]. Classification scores can also be merged with simple operators [10, 20]. The fusion problem can be seen as a classification task [22], in particular by using Bayesian classification [15] or support vector machines [7, 23]. More complex systems attempt to conduct both the classification and the fusion. Boosting algorithm [6] allows building a set of weak classifiers which outputs are fused. This idea is then withdrawn and extended to genetic algorithm [11]. We propose a fusion mechanism called hierarchical genetic fusion of possibilities (HGFP).

We assume that the output of an elementary classifier expresses the possibility of its associated class. For example, let $L_{water}(color)$ denote the SVM model that is trained on color ILSA signatures for the class water. Given a shot s , the output of $L_{water}(color)$ provides an information about the possibility to have the class water with respect to color features. Possibility logic is then used to achieve the fusion and we propose a hierarchical structure to model the fusion of many classifier outputs in this framework. Genetic algorithms are naturally introduced to find best fusion operators as well as the most appropriate hierarchical structures.

In the next section, we start by presenting possibility logic and fusion operators, and continue with the hierarchical structure and genetic algorithms.

3.1. Possibility Logic

It is not the scope of this paper to fully present the possibility logic. Thus, it is simplified to understand the basis necessary for our application. More information about the logic and its applications can be found in [4]. Here, we propose to do the fusion independently per class. The set of possible events is then a singleton $E = \{e\}$ that expresses the presence of a given class. We define the function $\pi : E \rightarrow [0, 1]$ that represents the state of knowledge distinguishing what is plausible from what is less plausible. π is indirectly provided by classifier outputs which values have to be mapped into $[0, 1]$. This can be done by using different functions, such as: min-max, Gaussian, sigmoid or exponential normalization.

The possibility logic defines the fusion as follows. Let E_1 and E_2 be two sets of possible events with their respective possibility distribution π_1 and π_2 . Let $E_{\oplus}$ the fusion of E_1 and E_2 with the operator $\oplus$. $E_{\oplus}$ is composed of:

1. $E_1 \cup E_2$ with $\pi_{\oplus}(e) = 1 - \oplus(1 - \pi_1(e), 1 - \pi_2(e))$ $e \in E_i$
2. $\{e \in E_1 \cap E_2\}$ with $\pi_{\oplus}(e) = 1 - \oplus(1 - \pi_1(e), 1 - \pi_2(e))$

Different fusion operators exist in this framework: minimum, maximum, t-norm, arithmetic mean, geometric mean, bounded sum, product and probabilistic sum. Each of them have different inference properties. Operators can be conjunctive (highest possibility is preserved), disjunctive (do not favor highest possibility), idempotent (redundancy is preserved) or reinforcement (redundancy is emphasized). Unfortunately, methods to select the right operator do not exist. Moreover, the fusion is conducted by combining two sets of events and when more than two sets are involved the fusion is lead iteratively. In order to find the most appropriate operators and fusion structure, i.e. the order that have to be used to fuse multiple sets, we model the fusion function as a binary tree which is build by a genetic algorithm as we see in the next part.

3.2. Binary trees and genetic algorithms

We presented in [18] a preliminary fusion method using binary trees and genetic algorithms. Here, the method is extended in the framework of possibility logic and also generalized to dynamic trees of variable sizes.

The main problem is to model the fusion function that was previously presented. The fusion function is a set of operations that are recursively applied to different set of

events. The order in which operators are applied is crucial since it refers to different fusion functions. The natural way to model such functions is therefore to use binary trees, whose leaves are the sets of events and internal nodes the fusion operators.

To summarize, the complete fusion chain firstly normalize classifier outputs in $[0, 1]$ thanks to a normalization function; next obtain possibility values are weighted by a priori possibilities; finally, these values are fused with respect to a binary tree and its associated fusion operators. The whole chain is depicted in the figure 3. The main question at his

Fig. 3. Proposed fusion function.

stage is to find the right tree structure with the appropriate attributes for leaves and internal nodes, the right normalization operators and the right a priori possibilities. Given the high number of possible configurations and the complexity of the fusion function, genetic algorithms provide a convenient tool to solve such a problem. Genetic algorithms are evolutionary processes that rely on the evolutionary ideas of natural selection and genetic transformations. The basic concept is to simulate the evolution of a population (potential solutions) by means of various genetic transformations (to create new solutions) and the selection of best elements with respect to a fitness function. In practice, we need to define the genetic transformations that can be applied on our potential solutions and the fitness function that quantifies the fitness of the solution to our problem. Typical transformations are the mutation of a single element and the crossover between two elements. Here follows the two genetic transformations specific to binary trees and a fitness function that we propose for our fusion problem.

The transformation on binary trees used for the experiments of this paper are based on Remy's identity. Remy's identity allows to uniformly expand or shrink a tree by selecting a random node and direction. The expansion proceeds as follows to obtain $n + 1$ leaves:

1. suppose we have a binary tree with k internal nodes, i.e. fusion operators, and $k + 1$ leaves, i.e. sets of events,

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 4),
4. repeat the process until the $n + 1$ leaves are in the tree.

A binary tree can be shrunk in a similar way using the opposite transformation.

1. select a random leaf ($\circ$),
2. remove the leaf,
3. replace its parent ($\star$) by its brother ($\diamond$), (figure 4).

Fig. 4. Remy's identity to add or remove a leaf.

On one hand, the mutation transformation that should transform a potential solution to another potential solution randomly modifies the tree structure including its size, picks up fusion operators for each internal node, attributes classifier outputs to all tree leaves, selects normalization functions and picks up a priori possibilities. On the other hand, the crossover transformation that should merge two potential solutions (a father and a mother) to a potentially better solution randomly selects a subtree of the mother that is grafted to the father and, selects weights, normalization functions and a priori probabilities that are copied into the father.

Genetic algorithms now require a fitness function to evaluate potential solution and keep the best ones. The focus of our application is information retrieval thus, fusion solutions are evaluated thanks to the mean precision values of the retrieval results. A training set is used for this purpose and solution fitnesss are evaluated on these data knowing the ground-truth, i.e. we know if a retrieved video shot is relevant or irrelevant. Let $D = \{(d_i, c_i)\}$ the set of training video shots and their class ordered with respect to the fusion score. The fitness or mean precision value is then:

$$\sum_i \frac{\text{number of relevant shots found}}{i = \text{number of retrieved shots}}$$

Fig. 5. Comparison of SVM-based fusion, our previous approach based on probabilities and the presented approach HGFP.

4. EXPERIMENTS

Experiments are conducted in the context of TRECVID 2005. Our fusion algorithm HGFP is evaluated on the task of high-level feature extraction which aims at ordering shots with respect to their relevance to a semantic class. Proposed semantic classes in 2005 are building, car, fire/explosion, U.S. flag, map, mountain, prisoner, sport, people walking/running and waterscape/waterfront. The quantitative evaluation is given by mean precision values of retrieval results limited to 2,000 retrieved shots. The training data set of TRECVID 2005 is composed of about 80 hours of news programs from American, Arabic and Chinese broadcasters. The set is split in three equal parts, chronologically by source, in order to train SVM models, find the best fusion function and evaluate our system performance.

Here, we present three comparisons. First, we compare our approach using fixed length trees with an SVM-based fusion method. Then, we compare fixed-length and variable-length trees. Finally, we introduce motion features and analyse their impact on final classification performances.

Figure 5 illustrates the retrieval performance per semantic class. The first bin gives random performance. The second bin gives performance when the fusion is achieved by SVM. The third bin is obtained with our previous fusion approach as presented in [18], only min, max, sum and product operators are used. Finally, the last bin is obtained with the presented approach. We notice the efficiency of the proposed approaches compared to SVM. One disappointing point with respect to average performance is that HGFP does not really outperform our previous approach. However, we obtain very similar performance while the space of

Fig. 6. Comparison of performances using fixed length tree and variable length trees.

potential fusion functions is bigger. The algorithm is then efficient in finding the right fusion function and it seems that previously available operators were already sufficient for our application. We point out high differences for classes with very few positive examples in our data set (classes mountain(6) and waterscale(10)). Indeed, in that case the training stage is not reliable since many possible configurations have the same performance and the selection is then done empirically.

Figure 6 compares performance obtained on fixed length trees and variable length trees. In the first case, all input features are used whereas the second case allows to discard some features and also to have redundant ones. Mutation and crossover genetic operators allow to create trees with different sizes. We notice that increasing the space of possible solutions does not imply that a better solution is found. We will have to investigate two potential sources of this behaviour. Either our genetic algorithm stops before the space of potential solutions is well explored or our algorithm is over-fitting the training data.

Figure 7 shows performance when motion features are introduced. Motion features are composed of motion histogram over the shot, mean camera motion and object motion histogram over the shot. It is important to notice that this feature has a very little impact on classes that are motion independent. It means that the proposed approach can deal with unreliable sources of information especially when using variable length trees. On the other hand, motion dependant features, such as sport(8) and people walking or running(9), have their performance highly improved.

To conclude this section, figure 8 gives examples of first retrieved shots on TRECVID 2005 dataset for the classes car, map and waterscape, to illustrate the efficiency of our

Fig. 7. Introduction of motion features.

classification method.

5. CONCLUSION

This paper extended a new fusion method of classifier outputs that we proposed in [18]. Even if performances are not really improved, the framework is well defined and the algorithm is more flexible. Classifier outputs are mapped into $[0, 1]$ in order to represent possibility values. Then, these values are combined thanks to fusion operators proposed in the possibility logic framework. The originality of our method is to automatically select most appropriate operators and the order in which they are applied. For this purpose, the fusion function is modeled by a binary tree. Then, genetic algorithms are used to find the best tree structure and associated operators.

The evaluation of our fusion method was conducted on TRECVID 2005 video database. We proposed to fuse four features that describe the visual content of video shots. Selected features are complementary since they include color and texture features extracted on homogeneous regions or salient regions. However, there is not any clue on how to combine them and a fusion algorithm is required to obtain the best performance. First, we compared an SVM based fusion method and our algorithms. Then, we added motion features to better describe shot content. Results are very promising on the difficult problem of video shot content detection.

We plan to investigate two different tracks. Firstly, we will have a deeper look at the genetic operators applied to our tree to find out why variable size trees do not better model the fusion function. Secondly, we will extend our work to multi-class problems. The idea is to take into account the correlation that exists between different semantic

Fig. 8. Examples of first retrieved shots for car, map and waterscape classes.

classes.

6. REFERENCES

- [1] Information technology - multimedia content description interface - part 5: Multimedia description schemes. ISO/IEC 15938-5, 2003.
- [2] Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines*, chapter Kernel-Induced Feature Spaces. Cambridge University Press, 2000.
- [3] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.
- [4] D. Dubois and H. Prade. Possibility theory and its applications: a retrospective and prospective view. volume 1, pages 5–11, 2003.
- [5] P. Felzenszwalb and D. Huttenlocher. Efficiently computing a good segmentation. In *Proceedings of IEEE CVPR*, pages 98–104, 1998.
- [6] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. pages 148–156, 1996.
- [7] G. Iyengar, H. Nock, C. Neti, and M. Franz. Semantic indexing of multimedia using audio, text and visual cues. In *Proceedings of ICME*, 2002.
- [8] T. Joachims. *Advances in Kernel Methods - Support Vector Learning*, chapter 11 (Making large-Scale SVM Learning Practical). MIT Press, 1999.
- [9] 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.
- [10] 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.
- [11] 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.
- [12] M.R. Naphade, T. Kristjansson, B. Frey, and T.S. Huang. Probabilistic multimedia objects (multijets): a novel approach to video indexing and retrieval. In *Proceedings of IEEE ICIP*, volume 3, pages 536–540, 1998.
- [13] Dymitr Ruta and Bogdan Gabrys. Classifier selection for majority voting. *Special issue of the journal of INFORMATION FUSION on Diversity in Multiple Classifier Systems*, 2004.
- [14] Nicu Sebe and Michael S. Lew. Salient points for content-based retrieval. In *BMVC*, 2001.
- [15] X. Shi and R. Manduchi. A study on bayes feature fusion for image classification. In *Proceedings of IEEE CVPR*, volume 8, june 2003.
- [16] Fabrice Souvannavong, Bernard Merialdo, and Benoit Huet. Video content modeling with latent semantic analysis. In *Third International Workshop on Content-Based Multimedia Indexing*, 2003.

- [17] 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.
- [18] Fabrice Souvannavong, Bernard Merialdo, and Benoit Huet. Multi-modal classifier fusion for video shot content retrieval. In *Proceedings of WIAMIS*, 2005.
- [19] TRECVID. Digital video retrieval at NIST. <http://www-nlpir.nist.gov/projects/trecvid/>.
- [20] 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.
- [21] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.
- [22] Patrick Verlinde, Gerard Chollet, and Marc Acheroy. Multi-modal identity verification using expert fusion. *Information Fusion*, 1(1):17–33, 2000.
- [23] Y. Wu, E. Y. Chang, K. C.-C. Chang, and John R. Smith. Optimal multimodal fusion for multimedia data analysis. In *Proceedings of ACM MM*, pages 572–579, 2004.