University of Paris 6 at TRECVID 2006: Forests of Fuzzy Decision Trees for High-Level Feature Extraction

Christophe Marsala
Université Pierre et Marie Curie-Paris6
CNRS UMR 7606, LIP6,
8 rue du Capitaine Scott,
Paris, F-75015, France
Christophe.Marsala@lip6.fr

Marcin Detyniecki
Université Pierre et Marie Curie-Paris6
CNRS UMR 7606, LIP6,
8 rue du Capitaine Scott,
Paris, F-75015, France
Marcin.Detyniecki@lip6.fr

Abstract

In this paper, we present the methodology we submitted to the NIST TRECVID'2006 evaluation. We participated in the High-level Feature Extraction task. Our approach is based on the use of a Forest of Fuzzy Decision Trees through the Salammbô software.

1 Structured Abstract - Summary

Here we present the contribution of the University of Paris 6 at TRECVID 2006 [1]. It concerns only the High-Level Feature Extraction task. The approach focuses on the use of a Forest of Fuzzy Decision Trees (FFDT) and is based on a rather simple image description.

In the following, we start with a short summary of the used method and starting from Section 3, our approach is detailed. First, we describe the particularities of our image descriptors. Then we explain how the training (Section 4) and classification (Section 5) was performed. Before concluding, the submitted runs are discussed in details (Section 6).

1.1 Brief Description of the Submitted Run

Here is the general information about the submitted run:

The task:	High-Level Feature Extraction.
Type:	A - system trained on TRECVID development collection data, and common annotation of such data.
Data used:	- XML files that provide the time-codes of each shot (Master shot references by [6]), - All the image files (the keyframes), - Annotations files for the devel keyframes.
Pre-treatment:	- Each keyframe was segmented into 5 regions (see Section 3.1), - An HSV histogram was computed for each region (see Section 3.1), - Temporal information about each shot was extracted from the XML files. (see Section 3.2).
Training:	A Forest of Fuzzy Decision Trees (FFDT) was constructed and trained from the Devel data set (see Section 4).
Ranking:	The Forest of Fuzzy Decision Trees (FFDT) was used to rank the shots from the Test data set (see Section 5.2).

1.2 Comments on the Run

1.2.1 Relative Contribution of each Component

Visual Information Descriptors are crucial since they are at the basis of the learning process. We choose to segment the keyframes into a set of rectangular regions and work on their color description. The choice of the number of regions and the number of bins for the histogram has still to be optimized. We believe that by doing so, we will manage to isolate important descriptors, thus helping the learning algorithm to focus on the discriminative variables. Moreover, more complementary visual descriptors should be added in order to enhance the possibilities of choice of the learning algorithm (FDT) for its decisions.

Video Information Descriptors are also as important. We chose to include temporal information brought by the shot's position in the video and its duration.

Training (Forest of Fuzzy Decision Trees) is the heart of our approach. The use of decision trees enables us to automatically discover the discriminating features. Moreover, the fuzzy logic theory provides a more robust treatment of numerical values of the descriptors. In fact, we have soft decisions avoiding any threshold effects.

Ranking using Forest of Fuzzy Decision Tree. Here again, the fuzzy logic theory implies a robustness when handling numerical values. Moreover, it enables us to obtain a degree for each feature, for each keyframe [5]. Since we have a Forest of Decision Trees, we obtain a set of degrees (decisions) for each keyframe. The final degree, used for the ranking, is an aggregation of the individual decisions. We believe that this "collegiate" decision provides a better coverage of the test space and is more accurate in terms of the degree, thus implying a better ranking.

1.2.2 Overall Analysis

In the previous TRECVID competition, we presented for the first time the use of Fuzzy Decision Trees for this kind of application [5]. The approach provided as result a set of classification rules which were human understandable, thus allowing further developments.

We discovered that, when we address large, unbalanced, multiclass data sets, a single classifier as the FDT is not sufficient. For instance, the space of negative examples is so large (proportionally to the positive examples) that we can not model it correctly. Based on this observation, we proposed in the 2006 challenge a Forest of FDTs, which is a way of covering better the whole input space. During the 2005 challenge we also noticed that a single FDT tends to provide the same degree of "certainty" for groups of keyframes. In fact, as a classification algorithm it optimizes the classification of all the examples and not the ranking of the results. Again the use of a Forest is a way to make a differentiation, and thus obtain a more accurate degree for the ranking, since every tree may vote differently.

The presented runs are an underestimation of what could be easily obtained. In fact, for time reasons we only submitted results with Forests of four FDT, for each of the 39 features. This is clearly an insufficient number and in Section 6 we present further results.

2 Introduction

The method used for the NIST TRECVID'2006 evaluation task is based on a Forest of Fuzzy Decision Trees (FFDT). More precisely, we used the Salammbô software, which was developed in our team at the Computer Science Department of the University of Paris 6 (LIP6).

A first preliminary step, before the construction and the use of a FFDT, consisted on transforming the data (devel and test set of the shots extracted from the video) in order to be processed by the Salammbô software.

Then, the main process is decomposed in three steps. In Section 3, the generation of vectors of descriptors from the keyframes and the XML files is presented. In Section 4, the training process, i.e. the constitution of training sets that should be process by the Salammbô software to construct FDT, is described. In Section 5, the method of processing FDT to classify keyframes is explained . In particular, we focus on the process of aggregation of the individual FDT decisions, thus enabling the ranking of the test keyframes. Before concluding, in Section 6 each of the performed runs is detailed.

3 Extraction of Image Descriptors

3.1 Visual Information Descriptors

The *Visual Information Descriptors* are obtained directly and exclusively from the keyframes. In order to obtain spatial-related information, we segmented the image into 5 regions (see Figure 1). Each of them corresponds to a spatial part of the keyframe: top, bottom, left, right, and middle. The five regions do not have the same size, which reflects the importance of the contained information based on its position.

Figure 1: Spatial segmentation of a Keyframe

Afterwards, for each region we computed the associated histogram in the HSV space. Based on the importance of the region, we choose to compute the histogram in a more or less precise way (i.e. number of bins): 6x3x3 for Middle, and Bottom, 4x3x3 for Right, and 4x2x2 for Left, and Top.

At the end of this procedure, we obtain the what we called the Visual Information Descriptors, a set of numerical values (belonging to [0,1]) that characterizes every keyframe.

3.2 Video Information Descriptors

All the *Video Information Descriptors* we used, was extracted from the master shot reference XML file associated to the video, which is nothing else than the result of a shot detection process (here by [6]).

For a given keyframe, these descriptors were extracted by parsing specific XML tags associated to each shot. In this way, for every keyframe we obtained the following information:

- the name of the keyframe and its kind: representative (RKF) or non representative (NRKF)
- the temporal position (timecode of the beginning) of the shot containing the keyframe
- the temporal position of the keyframe in the shot (timecode since the beginning of the shot)
- the duration of the shot containing the keyframe

At the end, we obtain a second set of numerical values that characterize the keyframe and the shot to which it belongs. This second set of information is called Video Information Descriptors.

3.3 Class Descriptor

The *Class Descriptor* is obtained from the human indexation of the video. It corresponds to the "correct" feature(s) to be detected on a shot. The Class Descriptor is extracted from the file obtained from the collaborative work of indexation of the devel video. Note that a keyframe can be associated with more than one class descriptor depending on the result of the indexation process.

4 Training with devel keyframes

In this section, we describe how, using the devel keyframes, the training enables us to obtain a classifier (FFDT) that will be used afterwards to classify and rank the test keyframes (see Section 5).

4.1 Building a training set

In order to use the Fuzzy Decision Trees (FDT) learning method, which is a supervised learning method, we must have a training set in which there are cases *with* the feature to be recognized and examples that do *not* possess that feature. In fact, decision trees construction methods are based on the hypothesis that the value for the class is equally distributed.

This hypothesis is not valid when considering the TRECVID'06 devel data set. For instance, for the Sports feature #1, in the whole devel set of indexed keyframes, there are 1570 keyframes with the Sports feature and 60066 keyframes without.

Thus, to have a valid training set for the construction of a FDT, we have to balance the number of keyframes of each class by (randomly) selecting a subset of the whole devel data set with an equal number of cases in each class.

4.2 Construction of a Forest of Fuzzy Decision Trees

4.2.1 Fuzzy Decision Trees

Inductive learning raises from the *particular* to the *general*. We build a tree from the root to the leaves, by successive partitioning the training set into subsets. Each partition is done by means of a test on an attribute and leads to the definition of a node of the tree. (For more details, see [5]).

The construction and the use of the FDT was done by means of the Salammbô software. This software was developed for building FDT efficiently and it enables us to test several kinds of parameters of the FDT [2]. Moreover, the automatic method to build a fuzzy partition on the set of values of the numerical attributes, mentioned above, was implemented [3] enabling us to avoid the prior definition of fuzzy values of attributes. Various parameters (t-norms, t-conorms) can be set in the Salammbô software and have been tested in the process of classification on different kinds of databases.

4.2.2 Forest of Fuzzy Decision Trees

The use of Forests of Fuzzy Decision Trees (FFDT) is crucial when we have large, unbalanced, multiclass data sets [4]. A forest is composed of a given number n of Fuzzy Decision Trees. Each FDT F_i of the forest is constructed from a training set T_i. Each training set T_i beeing a random sample of the whole training set, as described in Section 4.1. For instance, for the Sports feature, a subset of keyframes with each class (with Sports, or without Sports) was ramdomly selected in order to build a training set. In order to maximise the positive class, we decided to keep always the whole subset of keyframes with the Sport feature (1570 keyframes) and we just selected 1570 keyframes, randomly, within the subset of keyframes without the Sport feature.

Finally, for each of the 39 features, a Forest of FDTs was constructed.

5 Classification and ranking of test shots

5.1 Classifying keyframes with a Forest

The process of classification by means of a *single* Fuzzy Decision Tree was explained at the TRECVID 2005 challenge [5].

With a forest of n FDTs, corresponding to a single feature to be recognized, the classification of a keyframe k is done in two steps:

1. classification of the keyframes k by means of the n FDT of the forest: each k is classified by means of each FDT F_i in order to obtain a degree $d_i(k) \in [0,1]$ for the keyframe of having the feature. Thus, n degrees $d_i(k),\ i = 1...n$ are obtained from the forest for each k.

2. aggregation of the $d_i(k),\ i = 1...n$ degrees for each k in order to obtained a single value $d(k)$, which corresponds to the degree in which the forest believe that the k contains the feature.

The submitted run, $d_i(k)$ was valued by means of the Łukasiewicz t-norm (see [5]). The aggregation was done by summing the whole degrees: $d(k) = \sum_{i=1}^{n} d_i(k)$. Thus, $d(k)$ belongs to $[0, n]$. The higher $d(k)$, the higher it is believed that the k contains the corresponding feature.

5.2 Ranking Test shots

The ranking to be submitted for the challenge, is a ranking of Test *shots*. Thus, the degrees of all the keyframes $d(k)$ of a shot should be aggregated in order to obtain a single degree $D(S)$ for each shot.

One aggregating method is to add the degrees obtained for all the keyframes of the shot: $D(S) = \sum_{k \in S}(d(k))$. Another possible aggregation is considering that a shot contains a given feature if at least one of its keyframes (RKF or NRKF if any) contains the feature. Then, the degree $D(S)$ for the shot S containing the feature will be valued by: $D(S) = \max_{k \in S}(d(k))$.

As result, for every shot of the Test set, a degree is valued and all the shots can be ranked according to this degree. The higher $D(S)$, the higher it is believed that S contains the corresponding feature.

6 Experiments

In this part, we present results obtained by our method. First of all, we present the parameters that have been chosen for the submitted run. Afterwards, we present the parameters that have been chosen for further runs not submitted to the challenge but that leads to some improvements.

6.1 Submitted run

In our approach, presented in the previous part, there are several parameters that should be valued:

- the number of devel keyframes that should be used to construct a FDT. In the submission, at most 2500 keyframes with a feature have been selected for the training set. The training set is completed by as much keyframes without the feature than keyframes with the feature. We limited the size of a training set to 5000 keyframes.
- the operators used when classifying keyframes with a FDT (see Section 5.1). We used the Łukasiewicz operators.
- the number of Fuzzy Decision Trees in a Forest (see Section 4). We used only 4 trees.
- the aggregation of the degrees $d(k)$ of keyframes for a shot is done by summing: $D(S) = \sum_{k \in S}(d(k))$.

6.2 Improved runs

Further experiments have been conducted after the submission. In particular we focused on the size of Forest and we studied its impact on the results. We describe in the following parameters that were change with respect to the submitted run:

- the number of Fuzzy Decision Trees in a Forest (see Section 4). We used several size for the forest: 5, 11 and 20 trees.
- the aggregation of the degrees $d(k)$ of keyframes for a shot is done by considering that only a keyframe should contains the feature: $D(S) = \max_{k \in S}(d(k))$.

6.3 Results and Discussion

The results are presented in Table 1. For each graph, several runs of our method are presented and we compare also these runs with the median of the whole 88 results of submitted runs at TRECVID 2006 (only type A systems).

For the non-submitted runs, the values Inferred Average Precision (InfAP), and the number of hits at several depth (100, 1000, 2000) were computed by means of the `trec_eval` software[1] and the reference file `feature.qrels.tv06` that can be found on the website of TRECVID 2006 (directory for active participants). The median was extracted from the 88 submitted results (by all the methods) that were sent for evaluation to TRECVID.

The first observation is that by increasing the number of FDTs of the forest we improve the results and this independently of the measure. This confirms our hypothesis that FFDT is a suitable technique for covering large, unbalanced, multiclass data sets.

Secondly, we notice that our approach performs better (relatively to others) when looking at large recall list (e.g. rang 2000). Although the use of a Forest increases the accuracy of the values and the overall ranking, it seems that the improvement is still not uniform. Our belief is that classification algorithms do not optimize the ranking, but just the decision of the class.

Third, if we compare our approach to the others, and only by looking the classes where some reasonable result were obtained by all, we observe that our approach is particularly interesting for detecting the features 24-Military, 10-Desert, 30-Car and 17-Waterscape. Our approach compared to others is relatively weak for 3-Weather, 6-Meeting and 1-Sport. We guess that the reason for this is, in the one hand specialized systems for these features and, in the other hand, the simplicity of our visual descriptors. Further works will try to address this question.

[1] http://www-nlpir.nist.gov/projects/trecvid/trecvid.tools/trec_eval_video/

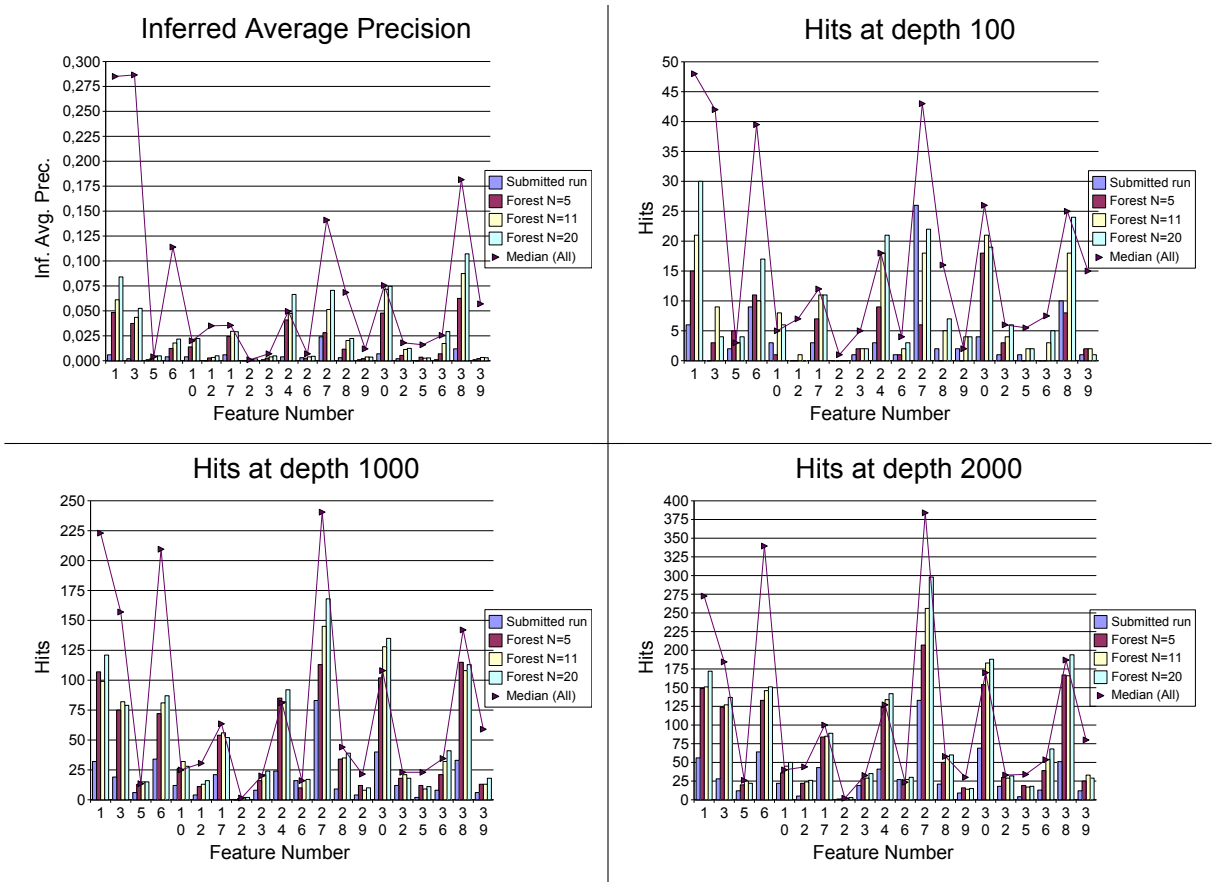

Table 1: Comparison of submitted run, improved runs (with several sizes of Forests), and median of the participants' runs at TRECVID'2006 (only type A systems)

7 Conclusion

Although, this work is at a preliminary stage, we obtained encouraging results. In the previous TRECVID competition, we presented for the first time the use of Fuzzy Decision Trees for this kind of application [5]. We discovered that, when we address large, unbalanced, multiclass data sets, a single classifier as the FDT is not sufficient. For instance, the space of negative examples is so large (proportionally to the positive examples) that we do not model it correctly. Based on this observation, we proposed in the 2006 challenge a Forest of FDTs, which is a way of covering better the whole input space.

During the 2005 challenge we also noticed that a single FDT tends to provide the same degree of "certainty" for groups of keyframes. In fact, as a classification algorithm it optimizes the classification of all the examples and not the ranking of the results. Again the use of a Forest is a way to make a differentiation, and thus a more accurate degree for the ranking, since every tree may vote differently.

As already stated in [5], one of the main drawbacks of our method is that it is based on very simple and generic visual descriptions. However, we should notice that in general decomposing the keyframes into regions improves the quality of the classification.

Finally, results of the use of Forest of Fuzzy Decision Trees seems very promising and we plan to study and adapt again the method in order to improve it for the high-level feature extraction process.

References

[1] Guidelines for the TRECVID 2006 evaluation - National Institute of Standards and Technology, 2006. http://www-nlpir.nist.gov/projects/tv2006/tv2006.html.

[2] C. Marsala. *Apprentissage inductif en présence de données imprécises : construction et utilisation d'arbres de décision flous.* Thèse de doctorat, Université Pierre et Marie Curie, Paris, France, Janvier 1998. Rapport LIP6 n^o 1998/014.

[3] C. Marsala and B. Bouchon-Meunier. Fuzzy partioning using mathematical morphology in a learning scheme. In *Proceedings of the 5th IEEE Int. Conf. on Fuzzy Systems*, volume 2, pages 1512–1517, New Orleans, USA, September 1996.

[4] C. Marsala and B. Bouchon-Meunier. Forest of fuzzy decision trees. In M. Mareš, R. Mesiar, V. Novák, J. Ramik, and A. Stupňanová, editors, *Proceedings of the Seventh International Fuzzy Systems Association World Congress*, volume 1, pages 369–374, Prague, Czech Republic, June 1997.

[5] C. Marsala and M. Detyniecki. University of Paris 6 at TRECVID 2005: High-level feature extraction. In *TREC Video Retrieval Evaluation Online Proceedings*, 2005. http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[6] C. Petersohn. Fraunhofer HHI at TRECVID 2004: Shot boundary detection system. Technical report, TREC Video Retrieval Evaluation Online Proceedings, TRECVID, 2004. URL: www-nlpir.nist.gov/projects/tvpubs/tvpapers04/fraunhofer.pdf.