UPMC-LIP6 at TrecVid'08: Balanced and Unbalanced Forests of Fuzzy Decision Trees for High-Level Feature Detection

Christophe Marsala, Marcin Detyniecki

Université Pierre et Marie Curie-Paris6
CNRS UMR 7606, LIP6,
104 avenue du Président Kennedy,
Paris, F-75016, France

{Christophe.Marsala,Marcin.Detyniecki}@lip6.fr

Abstract

In this paper, we present the methodology we applied in our submission to the NIST TRECVID'2008 evaluation. We participated in the High-level Feature Extraction task. Our approach is based on the use of a Forest of Fuzzy Decision Trees constructed by means of either a balanced sampling of the development data, or an unbalanced sampling of the development data.

1 Structured Abstract - Summary

Here we present the contribution of the University of Paris 6 at TRECVID 2008 [8]. It concerns only the High-Level Feature Extraction task. The approach focuses on the use of Forests 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 set of 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 only on common TRECVID development collection data: annotations and truth data publically available to all participants.
Data	- XML files that provide the time-codes of each shot (Master shot references by [7]), - All the devel image files (the devel keyframes), - Annotations files for the devel keyframes (provide by MCG-ICT-CAS[1])
Pre-treat.	- Each keyframe was segmented into 8 overlapping regions (see Section 3.1), - An HSV histogram was computed for each region (see Section 3.1),
Training	- A Forest of Fuzzy Decision Trees (FFDT) was constructed and trained from the Devel data set (see Section 4). - Two kinds of sampling was used in separate experiments: a balanced sampling of the development data, and an unbalanced sampling of the development data.
Ranking	- The Forest of Fuzzy Decision Trees (FFDT) was used to classify shots from the Test data set (see Section 5.2).

[1] *Multimedia Computing Group, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China.*

1.2 Comments on the Run

1.2.1 Relative Contribution of each Component

Visual Information Descriptors . We choose to segment the keyframes into a set of rectangular regions and work on their color description. By doing so, important descriptors could be isolated and, thus, could help 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.

Training (Forest of Fuzzy Decision Trees) . The use of decision trees enables us to automatically discover the discriminating features. In the Fuzzy Decision Trees, 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. Moreover, a set of FDT, a Forest of FDT (FFDT), is constructed to obtain a better classifier (a group of FDT performs better than a single one).

1.2.2 Overall Analysis

In the previous TRECVID competitions, we presented the use of Fuzzy Decision Trees for this kind of application [4]. The approach provided as result a set of classification rules which were human understandable, thus allowing further developments. This approach enables us to discover that, when addressing 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.

Thus, based on this observation, in [5] and [6], forests of FDT have been introduced to cover better the whole input space.

In the presented runs of this year, we introduced some novel methods to sample the devel set of keyframes in order to build the training sets that was used to construct the trees of the forests. This year, we introduced an unbalanced sampling in order to perform a comparison of the influence of the balance on the results. Moreover, we introduce this year the use of forests with a very high number of FDT (more than 100).

2 Introduction

The method used for the NIST TRECVID'2008 evaluation task is based on Forests of Fuzzy Decision Trees (FFDT).

A first preliminary step, before the construction and the use of a FFDT, consists in 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. 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 9 overlapping regions (see Figure 1). Each of them corresponds to a spatial part of the keyframe: corners (*"A"* regions), top and bottom (*"B"* regions), left and right regions (*"C"* regions), and central region (*"D"* region).

The regions do not possess the same size to reflect the importance of the contained information based on its position. Moreover, regions overlap in order to introduce a dependence between them.

For each region we computed the associated histogram in the HSV space. Based on the importance of the region, the histograms of each region is valued in a more or less precise way (i.e. number of bins): 4x3x3 (*"A"* to *"C"* regions)) or 8x3x3 for the central region.

Afterwards, the distance between the histograms of particular regions (*"A"* regions at the top, *"A"* regions at the bottom, and *"C"* regions in the middle) are valued in order to measure their difference.

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

3.2 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. The annotations of the Devel data set that have been used, have been provided by the Multimedia Computing Group, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China.

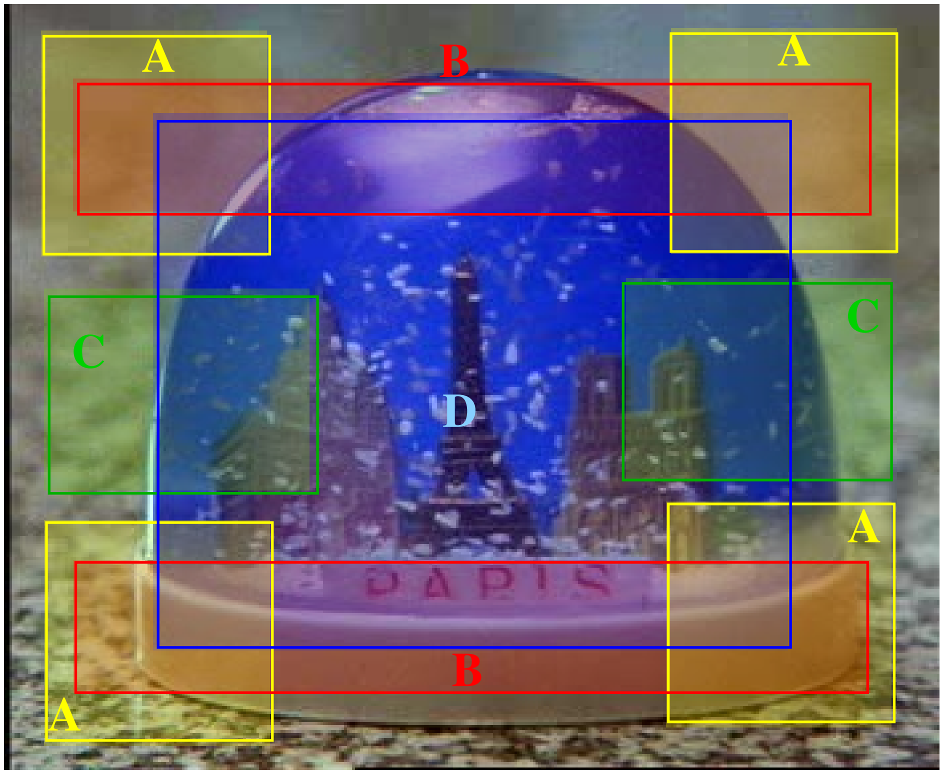

Figure 1: Spatial segmentation of a Keyframe

4 Training with devel keyframes

In this section, we recall briefly how the training enables us to obtain a classifier (FFDT) that will be used afterwards to classify and rank the test keyframes (see Section 5). For more details on the method, see [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.

To have a training set for the construction of a FDT, a sample of keyframes of each class is chosen by (randomly) selecting a subset of the whole devel data set. This year, we introduce two kind of sampling: a balanced one (as in previous works [6]) and an unbalanced one that is done by chosen more keyframes without the feature than keyframes with it. This process was conducted in order to have a training more close to the one involved in a bagging process [1, 2].

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 [4]).

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 [3]. Moreover, the automatic method to build a fuzzy partition on the set of values of the numerical attributes, mentioned above, was implemented 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 Forests of Fuzzy Decision Trees

A forest of FDTs was constructed for each of the high-level features. A FFDT 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.

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 has been explained in [4] and the process of classification by means of a forest of FDT has been explained in [5]. We recall here some basic steps of the method.

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.

This year, the aggregation has been done by summing the whole degrees: $d(k) = \sum_{i=1}^{n} d_i(k)$.

5.2 Ranking Test shots

The degrees of all the keyframes $d(k)$ of a shot are aggregated to obtain a global degree $D(S)$ for each shot to have the feature. The degree $D(S)$ for the shot S containing the feature is valued as $D(S) = \max_{k \in S}(d(k))$ (at least one keyframe of the shot should contain the feature).

As result, for every shot of the Test set, a degree is obtained from each FDT of the forest. The higher $D(S)$, the higher it is believed that S contains the corresponding feature.

Thus the shots are ranked by means of their degrees $D(S)$.

6 Experiments

In this part, we present the results obtained by our approach.

6.1 Submitted runs

The six runs that have been submitted this year are the following:

Run #1: Forest of 500 fuzzy decision trees. Combination of the forests from Runs #2, #3 and #6.

Run #2: Forest of 200 fuzzy decision trees. Unbalanced sampling. The aggregation of values in each FDT is done by means of the Zadeh's t-norms (see [4]).

Run #3: Forest of 200 fuzzy decision trees. Balanced sampling. The aggregation of values in each FDT is done by means of the Zadeh's t-norms (see [4]).

Run #4: Forest of 100 fuzzy decision trees. Unbalanced sampling. The aggregation of values in each FDT is done by means of the Zadeh's t-norms (see [4]).

Run #5: Forest of 100 fuzzy decision trees. Balanced sampling. The aggregation of values in each FDT is done by means of the Zadeh's t-norms (see [4]).

Run #6: Forest of 100 fuzzy decision trees. Balanced sampling. The aggregation of values in each FDT is done classically (see [4]).

All these comparisons can be highlighted in Table 1 and will be commented in the following.

6.2 Results

The results in Mean Inferred Average Precision for the six runs are presented in Table 1.

Run	#1	#2	#3	#4	#5	#6
mean IAP	0.024	0.0231	0.0193	0.0224	0.01805	0.0232

Table 1: Mean Inferred Average Precision for the runs

We recall that, among all the results (161 runs for type "A" algorithms): the average mean IAP is 0.0532, the median mean IAP is 0.0477, the mean IAP ranges from 0.0009 to 0.1667.

In the Figure 2, the detailled of the inferred average precision obtained for each run is shown. In this figure, *F500* stands for Run #1, *F200_unb* stands for Run #2, *F200_bal* stands for Run #3, *F100_unb* stands for Run #4, *F100_bal* stands for Run #5, and *F100_norm* stands for Run #6. The median of the Inferred Average Precision of all the results (161 runs for type "A" algorithms) for each feature is given.

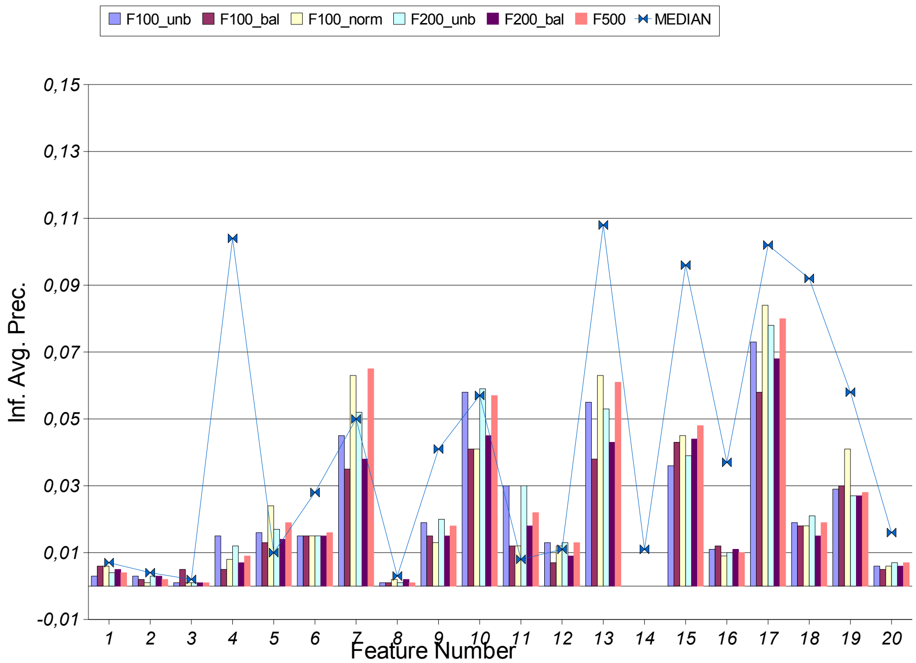

Figure 2: Detailed of the Inferred Average Precision

6.3 Discussion

As a global comment, we obtain the same kind of conclusion as in the previous years: our approach is generally better when finding good shots for a feature than when ranking them. It can be seen with the hits found within the 100, 1000, or 2000 shots of the ranking.

The use of unbalanced forest seems to be a good improvement and should be studied better by studying the influence of the ratio of unbalance. Moreover, the construction of heterogeneous forest that contains trees not constructed on the same approach performs some good results and should be studied better.

It should also be recalled that these results highly depend on the feature to recognize and a finest analysis of our results should also be done depending on that.

7 Conclusion

As already stated in our previous participation, one of the main drawbacks of our method is that it is based on very simple and generic visual descriptions. The main reason for that constant in our approach is that we focus our approach on the step after the generation of these descriptors. In further works, we will try new kind of descriptors in order to study their influence on the final results. However, we did not again focus on this part of the method, but we should do perhaps made something different in the future.

The obtained results this year highlights the importance of the sampling of the training set. The introduction of an unbalance of the classes, in order to render the reality of the development data set, brought out a gain in the results. One main reason for that seems to be linked to the fact that, by doing this, each fuzzy decision tree of the forest becomes more specialist in the recognition of the feature. Then, with this FDT, the prediction of the feature class tends to be less probable and tends to appear only when there are strong evidence that the shot contains the feature.

References

[1] L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

[2] L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

[3] 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.

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

[5] C. Marsala and M. Detyniecki. University of Paris 6 at TRECVID 2006: Forests of fuzzy decision trees for high-level feature extraction. In *TREC Video Retrieval Evaluation Online Proceedings*, November 2006. http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[6] C. Marsala, M. Detyniecki, N. Usunier, and M.-R. Amini. High-level feature detection with forests of fuzzy decision trees combined with the rankboost algorithm. In *TREC Video Retrieval Evaluation Online Proceedings*, 2007. http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[7] 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.

[8] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.