[FiSt99] *S. Fischer and R. Steinmetz;* Choosing efficient features for video classification; In SPIE Electronic Imaging 1999, San Jose, SPIE, January 1999

Choosing Efficient Feature Sets for Video Classification

Stephan Fischer[1], *Ralf Steinmetz*[1,2]

1
Industrial Process and System Communications
Department of Electrical Engineering and Information Technology
Technical University of Darmstadt
Merckstr. 25 • D-64283 Darmstadt • Germany
2
GMD IPSI
German National Research Center
for Information Technology
Dolivostr. 15 • D-64293 Darmstadt • Germany

email: {Stephan.Fischer, Ralf.Steinmetz}@kom.tu-darmstadt.de

Abstract

In this paper we address the problem of choosing appropriate features to describe the content of still pictures or video sequences including audio. As the computational analysis of these features is often time-consuming it is useful to identify a minimal set allowing for an automatic classification of some class or genre. Further it can be shown that deleting the coherence of the features characterizing some class is not suitable to guarantee an optimal classification result. The central question of the paper is thus *which* features should be selected and how they should be *weighted* to optimize a classification problem.

Keywords: Content processing, automatic video classification, video analysis, audio analysis.

1 Introduction

In the last few years the Internet and the World Wide Web have grown with an enormous speed, storing an incredible amount of multimedia data including video, audio and still images. As most of their sheer volume is unclassified, the need for content-based indexing and searching is beyond any question. Comparing the analysis methods described in literature, a large number of features is used to describe and analyze the content of video and audio. To classify still images a lot of different approaches can be found which often use color, texture and shape transformed to some sort of distance measurement to express image similarity [SCZ98, SJ98]. The task of video classification can be performed using a larger set of features including information on scene transitions (cuts, fades, dissolves, wipes) as well as camera (pans, wipes and zooms) and object motion. In the area of audio content classification features based on the analysis of loudness and frequency such as pitch and fundamental frequency are quite common. Also many other approaches extending the classification by text or speech recognition can be found [LS96, SC95].

A principal problem arises when classifying the content of an unknown video sequence automatically as often a single feature is not sufficient to obtain the desired reliability. Hence a feature set has to be used where the exact weights as well as the number of features to be used is unknown. A rather simple strategy is to enlarge the set until the classification is reliable enough risking that features are used which are equivalent and thus sometimes redundant. Taking into account that most features which can be used to analyze video and audio automatically need a considerable amount of machine power to be computed this approach is quite inefficient.

Our paper presents a systemaric method to select a feature set of minimal extent to still achieve an acceptable classification result but gain significantly in computation speed. This implies the decision which features should be used and how they

This work is supported in part by Volkswagen-Stiftung, Hannover, Germany.

should be weighted. For a given class we compute the distance measures "nearest neighbor", "inter class distance" and "inter and intra class distance" and select a minimal feature set which still yields acceptable classification results for a training set of images or video clips of the specific class by applying a sequential forward selection. These features can be used to find the class membership of an unknown sample to be tested without having to compute the whole features set. We also explain why other distance measures yield worse results and how our method can be completed by new distance measures which capture better the grouping properties of features than the currently known techniques do.

The paper is structured as follows: Following a review of related work, section 3 presents the syntactic and semantic features and their aggregation we used to classify different coherent groups of still images and film genres. Section 4 gives an overview of the distance measurements which can be used to calculate the density of our features in the sense of their ability to form clusters. In section 5 we explain how features should be selected to achieve the highest possible classification result at the lowest cost. In section 6 we provide the theoretic background of the factor analysis necessary for a computation of the coherence contribution. After having provided a prove of concept with our experiments in section 7 we conclude the paper in section 8 with an outlook.

2 Related Work

A lot of different approaches addressing the classification of digital video has been described in literature.

The recognition of various film genres has been described for example in [GSC+95, ZGST94]. The difference to our work is that the authors concentrate on one genre while we are able to identify a large number of genres. An issue we do not address is the use of icons like a logo to recognize genres.

The issue of image similarity and difference measures have been described in detail in various publications, for example in [Schal92, SCZ98, ZS94]. [SCZ98] use an approach similar to ours to classify still images.

A lot of other approaches omitting the classification of content and hence concentrating for example on text or speech recognition can be found in literature. However, it should be noted that it is not our intention to create a perfect classification system; we concentrate on the automatic classification of video *sequences*.

3 Syntactic and Semantic Features and their Aggregation

Features describing the content of audio and video can be separated into syntactical and semantie ones [FLE95]. *Syntactic features* can be extracted from digital video and audio without any baekground knowledge and hence describe the physical properties of the underlying content. Examples for syntactical features are eolor, edges, texture or audio frequency. A direct recognition of the content without any further transformation is in general impossible. Semantic or derived features allow for an interpretation of a video, for example for the automatic recognition of the genre of a movie. Examples for semantic features are the cut detection in the video domain or the calculation of the fundamental frequency in the audio domain.

The selection of features which have been used in the context of this paper can only be subjective as a huge number of features has been described in literature to classify the content of digital video and audio automatically. Research conducted to achieve an automatic abstracting of movies for example uses a different set of features [Ror93, PLFE96, LPE97]. However, the selection is limited as many features are highly eorrelated indicating an equivalence. Furthermore it is beyond the scope of this paper to define an optimal feature set to classify any kind of unknown material. The question to be answered is hence how to work with features already available.

In general two different kinds of features can be used:

- features which are calculated at a fixed point in time t
- features which are calculated aggregating a time interval.

As an example the video features *RGB-color* and *gray values* can only be calculated in single frames whereas motion vectors can only be estimated using a video sequence.

Syntactical as well as semantic features represent a transformation and an aggregation of the underlying digital film. To gain knowledge about the distribution of the original data we use the mean value, the median, the minimum and the maximum as well as the variance of the values contained within a time interval or at a specific point in time. For a still image, e.g., the mean, median, variance, minimum and maximum of the color values can be calculated.

We used the following features to analyze the content of video and audio:

- loudness and frequencies (syntactical features).

- pitch, fundamental frequency, onset, fast onset, offset and fast offset (semantics features).
- RGB-color, HSV-color, graytones, image similarity, edge distribution, Hausdorff-distance [Ruck94, MMZ95], number of objects in an image (syntactical features).
- detection of cuts, fades, dissolves, wipes and zooms (semantics features).

A problem is that the complexity of the features is different. While for example the color information can be computed per image (2D), the detection of cuts results in a 1D-number for a video segment. To be able to store the different complexity of the features we use arrays of different granularity. It will be shown in section 6 how the granularity should be chosen to avoid large arrays but also a too coarse quantization resulting in a data loss.

The result of the computation process of the training feature set of a given class is a set of arrays, one for each feature (subdivided into mean, maximal, minimal values, median and variance).

4 Distance Measurements

In the previous section we introduced the features we used for the automatic analysis of digital film. To derive higher semantics the following approaches can be used:

- features can be combined to derive semantics. A similarity measure together with a grouping of objects can for example be used to recognize objects.
- the distribution of the feature values can be used to classify the content automatically, for example to recognize the genre of a movie.

The issue to be addressed in this paper is the analysis of the feature distribution to derive semantics. The following difficulties can be identified when classifying a feature set:

- the amount of feature values. It is obvious that the mean color values calculated on a per frame basis cannot be compared for two movies which do not have an equal length.
- the feature correlation. It is possible that the use of two highly correlated features is redundant as the classification could be achieved already with one of the features.
- the number of features. A small feature set might not be sufficient to allow for a classification while a large set contains redundancy.
- the feature homogeneity. If features extracted from different movies are homogeneous they are obviously very important to recognize a movie genre.

It is the goal of our analysis to determine a feature set which suffices to solve a classification problem. This can be achieved either using the correlation and the factor analysis (see section 5) of the feature space or an estimation of the feature quality with regard to the classification.

To be able to analyze the quality of a specific feature with regard to the classification we used the nearest neighbor criterion, the Mahalanobis distance, probabilistic distances and Euclidean distances like the inter class distance and the inter- and intra class distance. These criteria and others are described in numerous publications [DK82]. The quality of a feature corresponds to the homogeneity of a feature class in this context. To guarantee a good classification performance a clustering has to be performed which is based on elements being as homogeneous as possible.

It is the goal of the Euclidean class measures to calculate distances between data clusters based on the Euclidean norm. Distances can be measured within a group (homogeneity of the group) or between data groups (separability of groups). Combinations of both measures take both aspects into account. In the following we consider the terms *group* and *class* as equivalent.

4.1 Inter Class Distance

The inter class distance measures the separability of classes quantitatively by estimating the pairwise Euclidean distance of the classes. If the inter class distance is large the classes are far apart from each other and thus separable. If a feature space where feature vectors are located which belong exactly to one of k classes and if each class k contains n_k vectors $\zeta^{ik}, 0 \le i \le n_i$ the inter class distance is defined as

$$k_{\text{inter}}(l) = \frac{1}{K-1} \sum_{k=1, k \neq l}^{l} \left[\frac{1}{n_k n_l} \sum_{i=1}^{n_l} \sum_{j=1}^{n_l} \delta(\xi^{jk}, \xi^{il}) \right]$$

where δ denotes a metric, e.g. the Euclidean metric. A disadvantage of the inter class distance is that it cannot be determined if classes overlap and how homogeneous they are. The inter class distance can hence not be used as a single criterion.

4.2 Inter and Intra Class Distance

Besides a large inter class distance the feature vectors of a class should be located in a close neighborhood in feature space. The inter class distance can therefore be extended to represent also the distances of the feature vectors inside a class. Using the notion of the inter class distance the intra class distance can be defined as

$$k_{\text{intra}}(l) = \frac{2}{(n_l - 1)n_l} \left(\sum_{i=1}^{n_l - 1} \sum_{j=i+1}^{n_l} \delta(\xi^{ik}, \xi^{jk}) \right).$$

An optimal distance measure should maximize the inter class distance while minimizing the intra class distance. Using the reverse inter class distance both values have to be maximized. The inter and intra class distance can hence be defined as

$$k_{\text{inter+intra}} = (1 - W) \cdot k_{\text{inter}} + W \frac{1}{k_{\text{intra}}}.$$

The weighting factor W adjusts the weight of the intra class distance. If the distance of the classes is the relevant criterion then W has to be chosen less than 0.5. It should be noted that the inter and intra class distance is not defined if k_{intra} equals zero. This case however cannot happen as then all feature vectors would be equal.

4.3 Nearest Neighbor Distance

The nearest neighbor criterion is a measurement for the homogeneity of a given class (e.g., newscast video). It counts, how many feature vectors x have a neighbor which is located in the same class like x. Formally the nearest neighbor criterion can be defined as

$$k_{nn} = \frac{1}{N} \sum_{i=1}^{N-1} nn(\xi^i)$$

with the function

$$nn(\xi) = \begin{matrix} 1, \text{ if class } \xi = \text{class}(NN(\xi)) \\ 0 \text{ else} \end{matrix}$$

In the notation we use N feature vectors ξ, a function class(ξ) representing the class of a vector and a function NN(ξ) identifying the nearest neighbor of the feature vector.

In addition to these a large number of distance measures has been described in literature, like the Mahalanobis distance or probabilistic distances, e.g. the Bhattacharyya distance [Fis97]. Experiments however showed that with regard to video content analysis the measures described above yield tbe most promising results [Fis97].

5 Computing the Feature Coherence using a Factor Analysis

An orthogonal approach to determine a minimal feature set is to use the factor analysis to eliminate the coherence of the features and to compute in such a way a minimal set of features which have a weight determined by their correlation. Less correlated features should get a higher weight than highly correlated indicators. It is the goal of the factor analysis to compute the contribution of the features to their co-variance. The idea behind the application of the factor analysis is that the use of highly redundant features should be avoided in terms of dropping one of two highly correlated features, preferably the one with the higher computational cost.

The principal approach of the factor analysis is the separation of a data matrix D into matrices A and F in a way that $D=AF$ [BE93]. In a first step the following transformation has to be applied:

$$z_{i,j} = \frac{d_{i,j} - \bar{d}_j}{s_j}, i \in [1, n], j \in [1, m],$$

where d denotes the n values of the m original data vectors and s_j represents the respective standard deviation. The main goal of the transformation is the standardization of the data. Furthermore $Z = A \quad F^T$ with $F^T = F$. The transformation enables a convenient processing of the correlations $c_{j,k}$ with

$$c_{j,k} = \frac{1}{n} \sum_{l=1}^{n} z_{j,l} z_{l,k}, k \in [1, m], l \in [1, m].$$

This is equivalent to

$$C = \frac{1}{n}(Z^T \quad Z)$$

where C is the correlation matrix. It follows

$$n \cdot C = Z^T \cdot Z = (a \cdot F^T)^T \cdot (A \cdot F^T) = (F^T)^T A^T A F^T = F A^T A F^T.$$

As $A^T A = n \cdot I$ we obtain $C = F \cdot F^T$. It follows that F does only depend on the co-variance matrix. To find F the co-variance matrix has to be transformed to a diagonal form using the eigen values. As C is symmetric and real m real eigen values can be computed. After scaling these to 1 the following equation is valid for the matrix V of the eigen vectors:

$$V^T C V = V^T F F^T V = (F^T V)^T (F^T V)$$

and with $X = F^T V$

$$V^T C V = diag(\lambda_1, ..., \lambda_m) = X^T X.$$

This can be used for the following equation:

$$F^T = X V^{-1} = X V^T \text{ and } F = V \cdot diag(\sqrt{\lambda_1}, ..., \sqrt{\lambda_m})$$

The square root of the k-th eigen value λ_k is thus proportional to the variance contribution of the k-th feature. Small variance contributions can be sorted out using the following criteria:

- only those entries k of the correlation matrix C are used for which λ_k is greater or equal to 1 [BE93].

- the number of dimensions is determined by the strongest decay within the order of the eigen values being sorted descending. Factors smaller than 0.5 will not be used.

The factor analysis is able to calculate the importance of features based on their correlation contribution. Note that the description above is simplified. The main component, main axis or image analysis can be found in [BE93].

6 Sequential Forward Selection for Suboptimal Classification

In the previous sections we described the theory which is necessary to estimate the quality of content analysis features. The quality is the premise for the derivation of weights for single features. Another issue which cannot be solved by the use of the quality is the feature *selection*. In most cases the deployment of the complete feature set is redundant as the classification problem could already be solved with a smaller set. Hence, the last step necessary to rate the features consists of a suboptimal selection of features. The optimal set is characterized by the fact that with no subset a classification with a lower error probability exists. A problem hereby is the determination of an optimal subset fulfilling these requirements. The following example illustrates the problem: using 300 features the task to identify a subset with 30 elements would require to evaluate

$$\binom{300}{30} \cong 1,7 \cdot 10^{41}$$

different subsets. In most cases this is not feasible. As a consequence a suboptimal subset has to be used which can be calculated for example using the Sequential Forward Selection (SFS) [Schal95]. SFS determines a subset of n elements being part of the whole set containing N features, following the constraint that the rating with a quality measure is maximal. The SFS algorithm is as follows:

1. Select the feature which has not been used so far and for which the quality is highest with regard to one of the distance measures. Mark this feature.

2. If the classification result using the set of selected features is satisfactory, stop and use selected feature(s) weighted with their quality to classify the specific class.
If not, go back to step 1.

7 Experiments

To apply the algorithms described above we looked into two different problems: the automatic recognition of film genres and the localization of commercials in a video stream. Both problems are classification problems as for the first application the genre of an unknown sample has to be classified while for the second a time interval of an unknown video sequence has to be compared with a pattern representing the commercial clip to be localized. In this section we explain the selection and the weighting process in detail and verify these results using the localization of commercials in video clips. A lot of other applications like the localization of audio events as well as the selection of key frames are possible using our approach. These have been described in [Fis97].

7.1 Automatic Recognition of Film Genres

In our experiments we used a total of 300 video clips containing 4000 frames each from different film genres (newscast, music, soccer, tennis, commercial, talkshow, cartoon, action movie, soap opera). We also used 20 clips (science fiction, soap opera, crime, action and cartoon) containing 20000 frames each to analyze if the performance depends on the length of the sequence.

To classify genres automatically we used the following audio features: loudness, frequency, pitch, fundamental frequency, onset, fast onset, offset, fast offset, frequency transitions and audio cuts. Onset and offset indicate the rise or decay of the audio signal while frequency transitions represent the glitch in time of the signal. In the video domain we used color, hue, saturation, gray values, image similarity, number of edge pixels per frame, Hausdorff distance of frames, optical flow distribution, number of segmented objects per frame, cuts, dissolves, wipes, fades and zoom. For each feature we computed the mean, median, maximum, minimum and variance. The result of our film analysis is thus a vector containing the analysis data.

To be able to classify the content of these segments the following problems have to be considered:

- the video clips can be of different length resulting in an unknown number of features to be compared. Comparing the feature *image similarity* being extracted of two clips with 20000 and 4000 frames, 20000 values have to be compared with 4000 to be able to determine a similarity of the two clips.
- the feature values are quite different making a correlation analysis difficult. *Color values* fall into the interval [0; 255] if an 8-bit representation is used while *image similarity* can be between 0 and 1.

To be able to compare video clips we normalized the feature values to the interval [0; 1] and stored the values in histograms. If the number of histogram entries is large the data can be transformed to a coarser histogram if necessary. If the histogram entries are divided by the total number of available entries for one feature the histograms can be compared and are thus independent of the length of the video clip. The result is a number of patterns which have to be calculated for the video clips being analyzed. Each pattern is an aggregation of the underlying audio and video material and includes all features that were computed in the form of a histogram.

When developing a classifier both training and test patterns have to be used. Training patterns are used to configure the classifier while test patterns allow for an estimation of the classifier reliability. The separation of the feature patterns into training and test pattern is thus an important decision. It is obvious that a huge number of patterns cannot be calculated using the complete set of available features as the computational cost is immense. However, numerous publications are available proposing algorithms to train and test the classifier [DK82], for example:

- resubstitution
- holdout
- leave-one-out
- rotation.

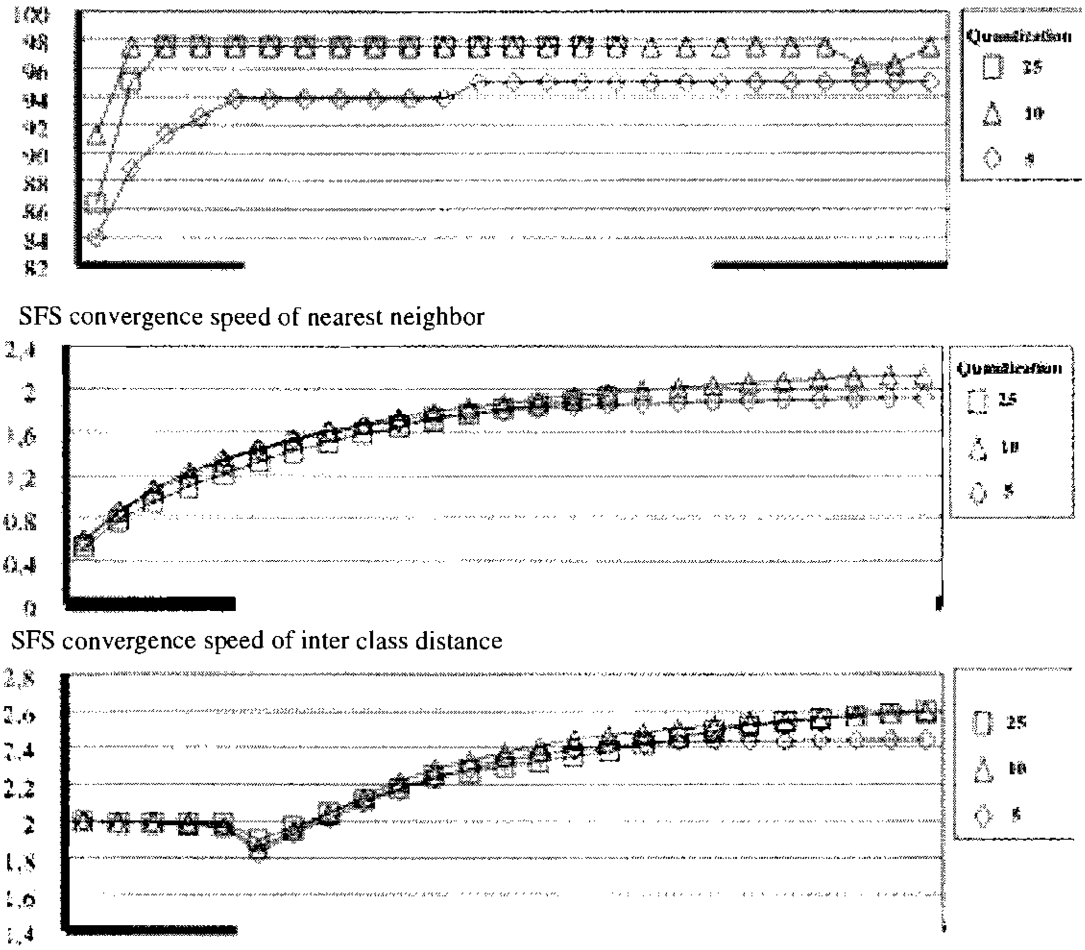
The resubstitution method uses identical test and training sets leading to an overestimated classification performance. The holdout method separates the feature set into two disjunct test and training sets leading to a possible underestimation of the classifier as not every pattern is used to train the system. The leave-one-out method uses only one pattern to train the system and the remaining patterns to test the classifier. After each step the training pattern is substituted with an unused training pattern. Although yielding the most precise results the computational cost of this approach must not be forgotten. The rotation method is a compromise between the leave-one-out and the holdout method separating the feature set into m subsets of v patterns where mv equals the total number of available patterns. After that the leave-one-out method is applied with the minor change that one subset is used to train the system while the other one serves for testing. After each step new subsets are chosen. To be able to gain a sufficient set for our experiment we used the rotation method rotating the training as well as the test set yielding to a total of 160.000 different sets.

7.1.1 Quality of Features

To identify the most efficient distance measure we compared the nearest neighbor criterion with the inter class distance and with the inter and intra class distance based on their quality as well as on the time necessary to compute the measures and on the convergence speed of the classification process. Unfortunately the quality is not comparable as the values of the distance measures are different. The nearest neighbor indicates in percent how many neighbors of a pattern are in the same class. The inter class distance as well as the inter and intra class distance measure the absolute distance between the patterns. Therefore these distances cannot be compared. We therefore evaluated the computing time as well as the SFS convergence speed to be able to examine the performance of the distance measures.

Each of the distance measures has been tested with the feature set quantized to histograms of different granularity. We used histogram widths of 50, 25, 10 and 5 bins to quantize the data. An entry of 1 at bin 50 of the feature "mean color" using a granularity of 50 would imply that 100 percent of the frames of the video segment had the highest possible color (white). In the following we tried to identify the number of features necessary for the construction of a profile which can be used as a basis for a comparison with unknown samples to be tested (for details on the construction of profiles see [FLE95]).

Looking at the convergence speed of the suboptimal classification process it turned out that the nearest neighbor criterion yields the best results for all of the examined classes. In most cases the desired recognition rate of 90 percent was reached after selecting only 3 features (of different nature depending on the specific class of the genre). It also turned out that the quantization of the arrays showed the best results at a length of 25. It seems to be clear that the rate should go down for a coarser quantization. In the opposite direction this seems to be surprising. With regard to the high dimensionality this effect is not surprising at all, as a very fine quantization yields also a high number of dimensions and the clustering gets worse. The convergence speed of the distance measures is shown in Figure 1.



SFS convergence speed of nearest neighbor

SFS convergence speed of inter class distance

SFS convergence speed of inter and intra class distance

Figure 1: SFS convergence of distance measures

We also compared the time necessary to compute the distance measures (see Table 1). The results indicated in Table 1 are normalized to be able to be compared using the value 100% for the inter class distance. As the nearest neighbor approach can be computed with the highest speed and as the convergence is faster than the other approaches we decided to use the nearest neighbor quality measure.

Distance measure	Computing time
Nearest neighbor	94,88%
Inter class distance	100%
Inter and intra class distance	133,23%

Table 1: Computing times of distance measures

7.1.2 Factor Analysis

First of all we used the factor analysis to erase the coherence of the features. The remaining adjusted features were then used to construct a profile on the basis of which samples can be classified using some distance measure. Most interestingly the recognition rate dropped significantly using the factor analysis. In comparison with the suboptimal selection described below the recognition rate was at least 10 percent (newscast), at most 27 percent and in the average 17 percent worse. This can be explained with the characteristic properties of the underlying class. As the factor analysis destroys this property, a decrease of the recognition rate could be expected. The recognition results are summarized in Table 2.

7.1.3 SFS-Recognition

In the following we tested the recognition rates of different classes of video clips using the nearest neighbor criterion to compute, which features should be used and how they should be weighted. We therefore used the SFS to identify the features which should be used to classify a genre and the quality measure of the nearest neighbor criterion to calculate the weights of the features per class. To recognize a genre we compared each feature vector with the profile for a genre using the following equation (the creation of profiles can be found in [Fis97, FLE95]):

$$\Delta = \sum_{i=1}^{N} \sum_{j=1}^{Q} (w_i \ |profile_{i,j} - pattern_{i,j}|),$$

where N denotes the number of features to be used and Q the quantization of the histograms. The weights w have to be calculated separately for each class using the method described above.

The recognition rates were the following:

Pattern class	Recognition rate using the nearest neighbor criterion	Recognition rate using the factor analysis
newscast	93,3%	80,71%
music clip	87,1%	60,11%
tennis	90,1%	72,83%
soccer	89,9%	68,93%
commercials	86,88%	61,36%
talkshow	91,2%	84,77%
cartoon	87,92%	77,9%

Table 2: Recognition results of automatic genre recognition.

A problem of this approach is that patterns where no profile is available cannot be classified. To circumvent this problem it has to be analyzed how similar pattern and profile have to be to identify a pattern. In 20 patterns with and without corresponding profiles were tested. It is obvious that a similarity of at least 30 percent is sufficient to identify the genre. All other patterns are classified as *unknown*.

To measure the recognition rate we created profiles consisting of 10 patterns each and tried to classify the remaining patterns. This was repeated 100 times with profiles created of different sets of patterns. The different recognition results can be explained with the different homogeneity of the pattern classes. Music clips and commercials are not that homogeneous as newscast leading to a lower performance.

The recognition rate is much worse for the factor analysis. An elimination based on a correlation analysis is thus impossible. This result is a hint for the existence of certain characteristics of the different classes which also includes a feature correlation. If the correlation is destroyed a reliable recognition becomes unreliable.

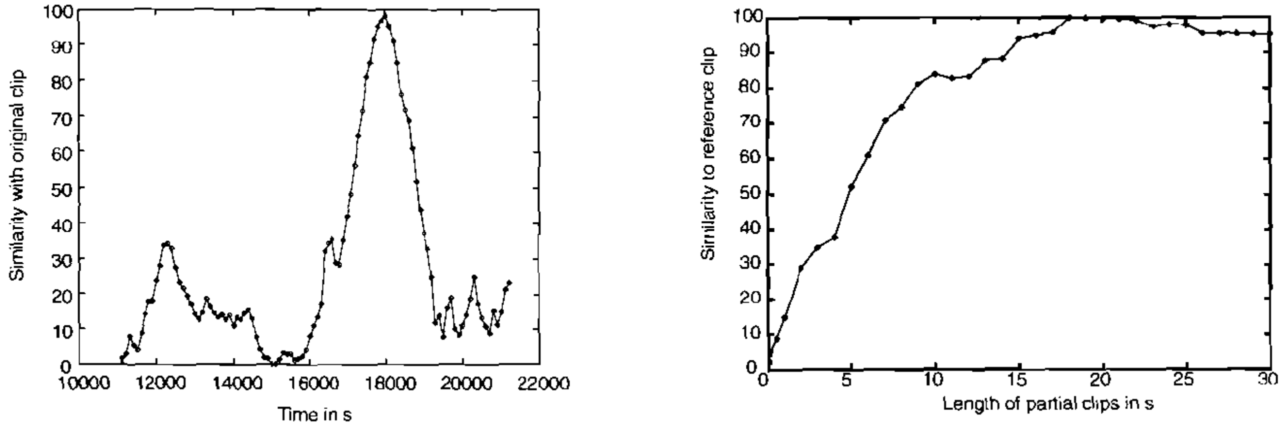
7.2 Localization of Commercials

Once a reliable approach to classify the content of a video clip is available other applications an be developed, for example the localization of commercials in a clip. Using a database of clips these can for example be cut out of a video. Another application is the control by the producing companies if a commercial has been broadcasted.

In our experiments we used a database of 16 commercials (each 160 s) characterized by the features described above. To localize them we used 5 clips of 30 min within which 5 of the database clips were contained. To localize a commercial we used the following algorithm:

1. calculate feature distribution of a time interval $[t_0, t_0 + 1]$ of length 1
2. calculate the Euclidean distance of the distribution to each pre-calculated distribution of the database
3. if the distribution is similar to a database distribution stop: clip recognized
4. if not, go on with the next time interval.

To be able to use the algorithm the length of the time interval has to be determined. To solve this problem we extracted small segments of different length from the original clips. Figure 2 (b) shows the similarity between the segments and the original clips. Obviously a length greater than 90 s is sufficient to describe the clip with a feature distribution. To compare the clips we used the features and the weights already computed when running the genre recognition for commercials. A localization of a commercial is shown in Figure 2 (a). It is clearly visible that the clip has been localized at t=18000.



(a) Localization of a commercial clip

(b) Necessary length of partial clips

Figure 2: Localization of commercials and necessary requirements

We also examined how the similarity changes if the test pattern is longer than the original pattern. Figure 2 (b) shows that the similarity is lower in that case but that a recognition is still possible. Table 3 summarizes our experimental results.

Clip number	Number of occurrences	Recognized	False positive
1	6	6	1
2	4	4	0
3	8	7	1
4	5	5	0
5	9	9	0

Table 3: Localization results of commercials

5 Conclusions and Outlook

In this paper we propose a new method for selecting and weighting features to analyze digital video and audio. We showed that this approach is well suited to recognize different groups of films in the sense of a classification. Surely, the main area of application of this idea is not the recognition of genres. Experiments showed, that the method can be applied to localize segments of films (e.g. detection of commercials) and to group these. Another application area is the classification of still images for which the method yields good results. Therefore the method can be used to search for certain instances of some type of image.

Certainly the method could yield better results in combination with pattern recognition techniques such as the detection of logos or the recognition of newscast speakers. It is the focus of our future work to examine these correspondences. As this is only an abstract the final paper will present a detailed overview of the experiments we conducted, as well as a mathematical background of the various distance measures and the factor analysis.

Acknowledgment

We would like to thank Wolfgang Effelsberg from the University of Mannheim and his group. This work is a result of many discussions and advice given by that group.

References

[BE93] K. Backhaus and B Erichson. *Multivariate Analysis Methods.* Springer Verlag, 1993.

[DK82] P.A. Devijver and J. Kittler. *Pattern recognition: a statistical approach.* Prentice-Hall, 1982.

[Fis97] S. Fischer. *Feature combination for content-based analysis of digital film.* PhD thesis, University of Mannheim, 1997.

[FLE95] S. Fischer, R. Lienhart and W. Effelsberg. *Automatic Genre Recognition.* Proc. ACM MM 1995, San Francisco, 1995.

[GSC+95] Y. Gong, L. T. Sin, C. H. Chuan, H. J. Zhang, and M. Sakauchi. *Automatic Parsing of TV Soccer Programs.* ICMCS95, 1995.

[LPE97] R. Lienhart, S. Pfeiffer, and W. Effelsberg. *Video Abstracting.* Communications of the ACM, 40(12), 1997.

[LS96] R. Lienhart and F. Stuber. *Automatic Text Recognition in Digital Videos.* Image and Video Processing IV, Proc. SPIE 2666-20, 1996.

[MMZ95] K. Mai, J. Miller, and R. Zabih. A Feature-based Algorithm for Detecting and Classifying Scene Breaks. Proc. ACM MM 1995, pp. 189-200, San Francisco, 1995.

[PLFE96] S. Pfeiffer, R. Lienhart, S. Fischer, and W. Effelsberg. *Abstracting Digital Movies Automatically.* Journal of Visual Communication and Image Representation, 7(4), 1996.

[Ror93] M.E. Rorvig. *A Method for automatically Abstracting Visual Documents.* Journal of the American Society for Information Science, 1993.

[Ruck94] W. Rucklidge. *Efficient computation of the mininum Hausdorff distance for visual recognition.* Dept. of computer science, Cornell University. TR-94-1454, September 1994.

[Schal92] R. J. Schalkoff. *Pattern Recognition: Statistical, Structural and Neural Approaches*. John Wiley and Sons, 1992.

[SC95] M.A. Smith and M. Christel. *Automating the Creation of a Digital Video Library*. Proc. ACM MM 1995, pp. 357-358, San Francisco, 1995.

[SJ98] S. Santini and R. Jain. *Beyond Query by Example*. Proc. ACM MM 1998, Bristol, UK, 1998.

[SCZ98] G. Sheikholeslami, W. Chang, and A. Zhang. *Semantic Clustering and Querying on Heterogeneous Features for Visual Data*. Proc. ACM MM 1998, Bristol, UK, 1998.

[ZS94] H. J. Zhang and S. W. Smoliar. *Developing power tools for video indexing and retrieval*. Proceedings SPIE Conf. on Storage and Retrieval for Image and Video Database, 1994.

[ZGST94] H. Zhang, Y. Gong, S.W. Smoliar, and S.Y. Tan. *Automatic Parsing of News Video*. Proceedings of IEEE Conf. on Multimedia Computing and Systems, 1994.