Integrating descriptions to characterize multimedia collections

CONFERENCE PAPER · JANUARY 2005

DOI: 10.1049/ic.2005.0727 · Source: IEEE Xplore

3 AUTHORS, INCLUDING:

Nicola Adami

Università degli Studi di Brescia

Riccardo Leonardi

Università degli Studi di Brescia

INTEGRATING DESCRIPTIONS TO CHARACTERIZE MULTIMEDIA COLLECTIONS

N. Adami, M. Corvaglia, R. Leonardi
DEA – University of Brescia
Via Branze 38, 25123, Brescia Italy – Fax +39 030 380014
{nicola.adami, marzia.corvaglia, riccardo.leonardi}@ing.unibs.it

Abstract

In this paper, an innovative approach is proposed for integration of descriptions regarding multimedia content collections. The proposed method has been applied to audio–visual content clusterings, which were extracted using different algorithms. With the support of low level features, obtained with further algorithms, all input clusterings are characterized and then merged in a new integrated clustering. The proposed method performs the integration taking into account the relative cluster size (granularity) and the element relationships among clusters. The experimental results show that the initial information of input clusterings is preserved in the resulting integrated clustering, both in terms of distribution and semantics. They also show that the amount of information available in the resulting clustering increases in terms of low level features.

1 Introduction

Research works on multimedia content analysis mainly focus on the extraction and characterization of digital documents at different semantic levels. Low level feature extraction (color, shape, spectrum, etc.) allows, for example, efficient retrieval and browsing of multimedia contents. Moreover, low level features are often jointly used to generate high level information with the intent to fill the "semantic gap" which represents the difference between automatically generated content description and the User expectations [6].

A relevant aspect of content analysis is the clustering of documents (or sub-segments of them) into groups according to one or more similarity criterion. This operation allows fast content browsing providing the user with a hierarchical representation of content relationships. Considering that different algorithms usually generate different clustering results, it is interesting to identify how these different outputs can be integrated. More in detail, given a generic multimedia content (a movie shot decomposition, a set of pictures, etc.) and given several algorithms for metadata extraction, the classification metadata provided by such algorithms for that content are compared and processed to obtain a structured, exhaustive and coherent description of all metadata available for the given content.

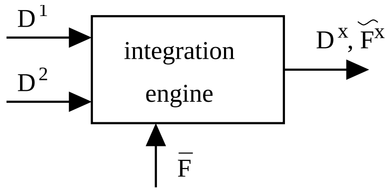

Figure 1: System for clustering integration.

A general approach for metadata integration was already proposed in a previous work for segment decomposition of videos [2]. In this paper, another relevant aspect of the content analysis is considered: the element clustering of multimedia collections. Since different algorithms usually generate different clustering results for the same collection, it is worthwhile identifying how these different outputs can be integrated. Usually, these outputs are at different semantic levels: some algorithms perform the clustering using a set of low level features, other ones interact with the user for the semantic characterization, etc. Hence, the descriptions provided by such algorithms for a certain content are compared and processed in order to obtain a structured, exhaustive and coherent description of all metadata available and also to decrease the semantic gap as much as possible.

For this purpose, in this work the cluster distributions D^1 and D^2 (given by clustering methods M^1 and M^2), with the support of further low level features $\overline{F}$, are processed to obtain an integrated cluster distribution D^x and the corresponding features $\widetilde{F^x}$ (Figure 1).

The paper is organized as follows. A preliminary cluster characterization, in terms of granularity, structure and low level features, is shown in Section 2. Therefore, a method for clustering integration at different levels of granularity is presented in Section 3. Finally, after some experimental results (Section 4), conclusions and future works are reported (Section 5).

2 Clusterings characterization

Let's suppose to apply a certain clustering method M to a set of audio–visual elements, for instance shots, images or audio clips. The result consists of a set of clusters whose elements have similar features, according to the used clustering method. Due to the fact that the clustering algorithm is not known, we need to extract some features to characterize each clustering as a whole. Thus, we analyze how each cluster can be described by using additional information extracted from features associated to cluster elements.

2.1 General aspects of clustering

A cluster method M produces a *Cluster Distribution* D usually with a tree structure of clusters. Some examples are reported in Figure 2. From the cluster distribution, some parameters can be extracted.

- *Range* — Each cluster consists of a set of elements. The range is the gap between the minimum number of elements in a cluster (L_{min}) and the maximum one (L_{max}).

- *Granularity Factor* — The granularity indicator g is given by the mean number of elements of clusters:

$$g = \frac{N_e}{N_c} \tag{1}$$

  where N_e is the total number of elements and N_c the total number of clusters. For example, if $N_e = 251$ and $N_c = 10$, the resulting granularity factor is $g = 25.1$. Or if N_e is the same and N_c is set to 45, the resulting granularity factor is $g = 5.58$. In general, high granularity factors ($g = 25.1$) correspond to distributions with a small number of clusters (Figure 2 a.) while low granularity factors ($g = 5.58$) correspond to distributions with a big number of clusters (Figure 2 b.).

- *Granularity Level* — Each granularity factor corresponds to a specific level of granularity. For instance $g = 25.1$ can be associated to granularity level $l = 0$ (Figure 2 a.) while $g = 5.58$ to granularity level $l = 1$ (Figure 2 b.). In general, high granularity factors corresponds to low granularity levels and viceversa. The granularity level, with the corresponding granularity factor, provides information about clusterings comparability.

  A set of elements can be clustered at different levels, as shown in Figure 2 c. Mainly, if the number of granularity levels is N_l, the parameter l assumes the values if the range $l \in [0, N_l - 1]$.

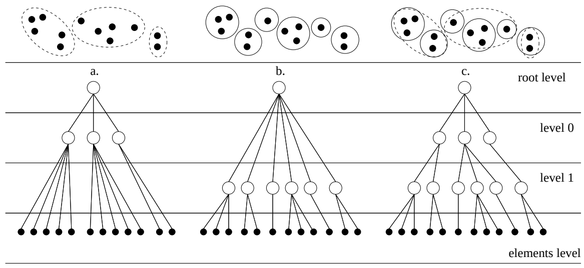

Figure 2: Examples of Cluster Distributions.

2.2 Cluster characterization

Given a clustering distribution D, obtained with a given clustering method M, each $i{-}th$ cluster C_i ($i \in [0, N_c - 1]$) of such distribution, at granularity level l ($l \in [0, N_l - 1]$), can be characterized mainly in two ways: by features associated to the entire cluster (*cluster features*) or/and by features associated to all the elements of the cluster (*elements features*). Usually, the first ones are high level features (semantics, text annotation, classification, etc.) while the second ones are low level features (color, shape, audio spectrum, etc.). Moreover, the cluster features are strongly jointed to the cluster distribution; in other words, they are included in the inputs D^1 and D^2 of the system shown in Figure 1. On the other hand, the element features are not necessarily connected to the distributions D^1 and D^2 because each element of the cluster is characterized by its own feature: in the most of the cases, the element features represent a separate input ($\overline{F}$) in respect to distributions D^1 and D^2. In this treatment, we consider the following scenario: the cluster features consist of text annotation, that is, each cluster C_i is characterized by semantic feature S_i, while the element features consist of low level features, that is each element of each cluster E_{ij} is characterized by a low level feature F_{ij}, with $j \in [0, N_i - 1]$ where N_i is the number of elements of cluster i.

As the purpose of this work is the clustering integration, all element features F_{ij} of a certain cluster can be merged together in order to better characterize the cluster, or, in other terms, in order to have another cluster features ($\widetilde{F_i}$) in addition to the semantic ones (S_i).

In an audio-visual context, low level features refer to color, pattern and so on. In order to optimize the low level integration, we should consider all possible types of low level features and then define the optimum integration method for each one on them. This approach has a couple of risks: it is difficult to implement and the effort could not produce a real gain. So, the most intuitive approach is the computation of the center of mass and the variance of all features F_{ij} of the elements E_{ij}, for each cluster C_i. The center of mass provides the characteristic low level features of the considered cluster C_i while the variance provides an indicator of the reliability of the obtained features for such cluster.

$$\widetilde{F}_i = \hat{\mu} = \frac{1}{N_i} \sum_{j=0}^{N_i-1} F_{ij} \qquad \sigma^2 = \frac{1}{N_i} \sum_{j=0}^{N_i-1} (F_{ij} - \hat{\mu})^2 \quad (2)$$

3 Clusterings integration

For clustering integration, we suppose to have two methods, M^1 and M^2, that generate two different cluster distributions D^1 and D^2 of the same set of audio–visual contents. The integration engine merges D^1 and D^2, with the support of additional features $\overline{F}$, and then creates an integrated distribution D^x with the corresponding features $\widetilde{F^x}$ (Figure 1).

The proposed method for clustering integration processes cluster distributions D^1 and D^2 at only one granularity level, such as the two shown in Figure 2 a. and b. This hypothesis does not mean that the output distribution D^x is at one granularity level: the resulting distribution can be at one granularity level (Figure 2 a. and b.) or at two granularity levels (Figure 2 c.). Therefore, we need a method to fix granularity level factors of D^x, given the two input distributions D^1 and D^2.

In the integration algorithm, the low level features play a fundamental role because they represent the discriminant factor of cluster integration. On the other hand, the original cluster distributions, provided by methods M^1 and M^2, should be preserved as much as possible. Hence, the integration should be performed at cluster level, not at elements level, with the support of low level features. If the integration is performed only at elements level, we could lose information provided by M^1 and M^2 and, from a practical point of view, we would simply apply a new clustering method, which are presumably different from the features used by methods M^1 and M^2.

In the clustering characterization of Section 2.2, we analyzed that, in addition to semantic features S_i already available for each cluster C_i, the elements features F_{ij} are combined, using equations (2), to characterize, with $\widetilde{F}_i$, each cluster C_i at cluster level. It follows that the integration is performed, where it is possible, using $\widetilde{F}_i$ in order to preserve the clustering information provided by M^1 and M^2 (semantics of each cluster S_i, distribution D, etc.).

The developed system is shown in Figure 3. It is totally MPEG–7 compliant. Each part of the system is explained in the following Sections.

3.1 Granularity evaluation

Before integrating cluster distributions D^1 and D^2, an evaluation of their relative granularity is required. Essentially, we need to understand if D^1 and D^2 belong to the same level of granularity. For this purpose a threshold is defined as:

$$th = \frac{max\{L^1_{min}, L^2_{min}\} + min\{L^1_{max}, L^2_{max}\}}{2} \quad (3)$$

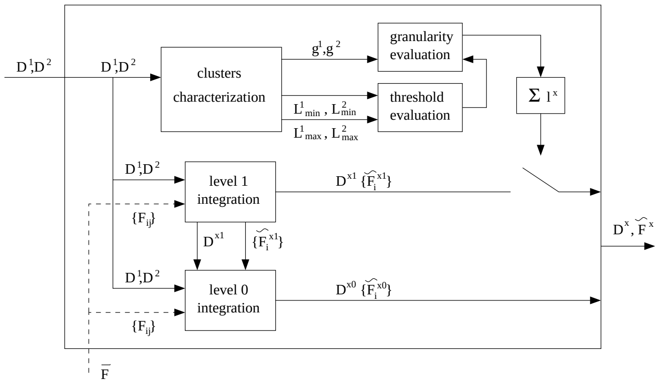

Figure 3: The developed system for clustering integration.

In our hypothesis, both D^1 and D^2 are characterized by one granularity level, so only one threshold is required. If the total number of elements is N_e and the number of clusters is N_c^1 and N_c^2, respectively for D^1 and D^2, the granularity factors g^1 and g^2 can be calculated with (1). Comparing them to threshold th, two significant situations can occur.

- Both g^1 and g^2 are under or above threshold, which means that D^1 and D^2 have clusters at the same granularity level. In this case, the integrated distribution D^x has one level of granularity ($l^x = 0$).
- The granularity factor g^1 is under threshold and g^2 is above threshold or viceversa. This means that D^1 and D^2 are at different level of granularity. The integrated distribution D^x has two levels of granularity ($l^x = 0$ and $l^x = 1$).

Figure 3 shows how the introduced parameters act in the proposed system. The cluster characterization block provides the *Granularity Factors* (g^1 and g^2) to the granularity evaluation block and the *Range* parameters (L^1_{min}, L^2_{min}, L^1_{max}, L^2_{max}) to the threshold evaluation block. The granularity evaluation block may generate one *Granularity level* parameter ($l^x = 0$) or two parameters ($l^x = 0$ and $l^x = 1$). Once the integration blocks generate D^{x1} and D^{x0} (next Section), the resulting granularity level parameters l_x are used to analyze if the distribution D^{x1} is valid or not: D^{x1} is valid only if $\sum l^x > 0$ (l^x assumes two values), otherwise, it is ignored and only D^x_0 is delivered as output.

3.2 Integration method

Given D^1 and D^2 at one level of granularity (hypothesis), we consider the worse case: the integration creates a merged cluster distribution D^x at two levels of granularity (for instance, Figure 2 c.). A merged distribution D^x at one level of granularity is a debauched case, as explained in previous Section.

In the proposed approach, the integration is performed in two steps: level 1 integration and level 0 integration

(Figure 3). The first one generates D^{x1} and the corresponding features $\{\widetilde{F_i^{x1}}\}$. Later, using D^{x1} and $\{\widetilde{F_i^{x1}}\}$, integration at level 0 is performed to obtain D^{x0} and $\{\widetilde{F_i^{x0}}\}$. If l^x assumes two values, D^{x1} and D^{x0}, with the corresponding features, represent the integrated cluster distribution D^x, with the associated features $\{\widetilde{F_i^x}\}$.

3.2.1 Integration at level 1 ($l^x = 1$)

At first level of granularity, the integration is performed by extracting common sub–clusters from D^1 and D^2: $D_1^x = D^1 \cap D^2$. For instance, if clusters $C_1^1 = \{E_1, E_2, E_{10}, E_{11}\}$ and $C_2^1 = \{E_3\}$ belong to D^1, while clusters $C_1^2 = \{E_1, E_2, E_3\}$ and $C_2^2 = \{E_{10}, E_{11}\}$ belong to D^2, the integrated D^{x1} consists of three clusters: $C_1^{x1} = \{E_1, E_2\}$, $C_2^{x1} = \{E_{10}, E_{11}\}$, $C_3^{x1} = \{E_3\}$. We can observe that sub–cluster C_2^{x1} corresponds to cluster C_2^2. This information suggests that the D^2 corresponds to lowest granularity factor and also that the original cluster distribution D^2 is widely preserved in D^{x1}.

To characterized D^{x1} clusters, equations (2) are applied and $\{\widetilde{F_i^{x1}}\}$ are generated.

3.2.2 Integration at level 0 ($l^x = 0$)

Suppose that distribution D^1 has the highest granularity factor. The integration at level 0 is performed in some steps.

- Using equations (2) all clusters of D^1 and D^2 are characterized through $\{\widetilde{F_i^1}\}$ and $\{\widetilde{F_i^2}\}$.
- All N_c^1 clusters of D^1 and all N_c^2 clusters of D^2 are mixed together: $D^3 = D^1 \cup D^2$. The obtained distribution D^3 consists of $N_c^3 = N_c^1 + N_c^2$ clusters which are at different levels of granularity. Therefore, a processing is required in order to prune clusters at lower granularity level.
- A probability is associated to each cluster C_i^3 of distribution D^3. Each cluster C_i^{x1} of D^{x1} (distribution at level 1) is certainly a sub–cluster of two clusters of D^3: one originally belonged to D^1 and the other one to D^2. For each cluster C_i^{x1}, two distance are calculated, using the distance measure EMD [11]:

$$d^1 = emd\left[\{\widetilde{F_i^{x1}}\}, \{\widetilde{F_i^1}\}\right]$$

$$d^2 = emd\left[\{\widetilde{F_i^{x1}}\}, \{\widetilde{F_i^2}\}\right]$$

  where $\{\widetilde{F_i^1}\}$ and $\{\widetilde{F_i^2}\}$ are the features originally associated to D^1 and D^2, while $\{\widetilde{F_i^{x1}}\}$ are the features associated to D^{x1}. Hence, the probability p_i^3 of each cluster C_i^3 is given by:

$$p_i^3 = \frac{1}{N_{si}^3} \sum_{k=0}^{N_{si}^3} s_k$$

$$s_k = \begin{cases} 1 & if \quad d^1 \leq d^2 \text{ and } C_i^3 \in D^1 \\ 1 & if \quad d^1 \geq d^2 \text{ and } C_i^3 \in D^2 \\ 0 & otherwise \end{cases}$$

  where N_{si}^3 is the number of sub–clusters of C_i^3. We observe that the probability p_i^3 is equal to zero when the cluster C_i^3 is equal, in terms of elements, to a cluster C_i^{x1} at level 1.
- Pruning operation consists of the elimination of clusters with probability p_i^3 under a chosen threshold p_{th} and with a number of elements smaller than the value given by $max\{L_{min}^1, L_{min}^2\}$.
- The previous step draws an approximation of the integrated distribution D^{x0}. It is an approximation because some sub–cluster C_i^{x1} (cluster at level 1) are not included anywhere while some others are included twice in clusters C_3^i. Consequently, a final redistribution of such clusters is required. If C_i^{x1} is absent, two situation can occur: first, if the number of elements is more than $max\{L_{min}^1, L_{min}^2\}$, cluster C_i^{x1} becomes a new cluster of D^3, with its low level features; otherwise, C_i^{x1} is merged with cluster C_i^3 at minimum distance between respective low level features (C_i^3 maintains its low level features). While, if C_i^{x1} is present in two clusters of D^3, C_i^{x1} is removed from cluster C_i^3 at maximum distance and C_i^3 preserves its low level features.

Finally, The resulting distribution D^3 is the integrated distribution D^{x0}, with the the original cluster features S_i and elements features $\widetilde{F_i^x}$. We observe that the semantic information and low level features do not change during the integration process, even if some sub–cluster are added or removed from the clusters.

4 Experimental Results

The considered content for testing is a video news from Portuguese TV, which is a video called "Jornal da noite" [10]. Its first 25000 frames (about 17 minutes) have been decomposed in shots using algorithm [3]; the resulting number of shots is 163. For each shot, one keyframe has been extracted with the same method used in [1]. Finally, for each keyframe three different types of visual-low level descriptors have been extracted with MPEG–7 eXperimental Model [9]: `EdgeHistogram`, `RegionShape`, `ScalableColor` [7]. These low level descriptors correspond to the input features $\overline{F}$ of the developed integration system (Figure 3). They are used by the system to merge the input clustering distribution D^1 and D^2.

Afterwards, three different clusterings method are considered.

- Clustering made by hand (M^1): each shot is classified according to a set of classes chosen by the user (each class is characterized by its own semantics).

Method	Distribution	N_c	L_{min}	L_{max}	g
M^1	D^1	11	4	25	14.82
M^2	D^2	44	1	12	3.7
M^3	D^3	23	1	25	7.09
M^3	D^4	54	1	8	3.01

Table 1: Characterization of cluster distributions.

- Clustering method M^2 obtained with algorithm [4]: given a set of pre–annotated shots, a clustering is performed using the vector quantization.
- Clustering method M^3 obtained with algorithm [5]: the user can choose the clustering criterion among a wide set of pre–computed MPEG–7 descriptors and, then, he/she can annotate each obtained cluster through the interface. For testing, we considered two separate criteria:

  1. `CameraMotion` [7] and `AudioSpectrumCentroid` [8];
  2. `ScalableColor` and `DominantColor` [7].

  In the fist case, we considered different types of features compared with the input features $\overline{F}$ (`EdgeHistogram`, `RegionShape`, `ScalableColor`), in order to test the generality of the proposed integration method. In the second case, we considered visual features in order to test if the proposed method provides better performances with features of the same type.

If we apply these three clustering methods to the set of shots previously obtained, we achieve the cluster distributions D^1 for method M^1, D^2 for M^2, D^3 and D^4 for M^3 (D^3 is obtained by applying criterion 1 while D^4 by applying criterion 2). Their cluster characterization is summarized in Table 1, where N_c is the number of clusters, $[L_{min}, L_{max}]$ the range of elements for clusters and g the granularity factor. Besides, each considered clustering method provides a semantic description S_i for each cluster: with method M^1, each cluster is classified according to the classes chosen by the user A (Table 4); method M^2 maintains the original shot semantics (Table 5); the interface of method M^3 allows the user to annotate each cluster (Table 6 for user B and criterion 1 and Table 7 for user 3 and criterion C).

The integration algorithm has been tested for some couple of clustering methods: M^1 with M^2, M^1 with M^3, M^1 with M^4, M^2 with M^3, M^2 with M^4. The probability threshold for integration at level 0 is set $p_{th} = 0.7$. In Table 2 the results are reported (N_c^0 is the number of cluster at level 0 while N_c^1 the number of cluster at level 1).

First, we can observe that the integration $int(D^2, D^3)$ is the only one at one granularity level. Then, the number of the clusters N_c^0 of $int(D^2, D^3)$ is approximately the mean of the number of clusters of the input distributions D^2 and D^3. Second, the other integrations $int(D^1, D^2)$, $int(D^1, D^3)$, $int(D^1, D^4)$, $int(D^2, D^4)$ are at two granularity levels. In this case, the number of clusters at level 1 (N_c^1), for each integration, is always the same regardless of the input feature $\overline{F}$ because the integration al level 1 consists of an intersection among clusters. The number of clusters at level 0 (N_c^0) is approximately very close to the number of the clusters (N_c) of the distribution at higher granularity. For instance, the number of clusters of $int(D^1, D^2)$ at level 0 is $N_c^0 = 17$ (Table 2) which is closer to $N_c = 11$ of D^1 than $N_c = 44$ of D^2 (Table 1). This result suggests that the cluster distribution at higher granularity level provides the most contribution in the integration process.

To further validate the obtained results, let's analyze the semantics. The considered audio–visual content for testing is a video news. The "anchorman" is a common label of the clusters for all distributions (Table 4, 5, 6 and 7). It represents a semantic feature. To evaluate if the proposed system preserves the semantics or slightly increase it, we consider all clusters with the semantic element "anchorman", for all cluster distributions integrated with D^1 and for `ScalableColor` descriptor as external feature $\overline{F}$. From Table 3, we can note that for the integrated distribution $int(D^1, D^2)$, the semantic information increases because all the elements *anchor:anchor_front*, *anchor:anchor_front_close*, etc. correspond to the sub–clusters. Instead, for $int(D^1, D^3)$ and $int(D^1, D^4)$, the semantic information increases but not in a proper way because many terms are semantically different from "anchorman". On the one hand, both input cluster distributions D^3 and D^4 are not accurate (Table 6 and Table 7), so there is a kind of wrong semantic propagation. On the other hand, the number of wrong semantic terms obtained in $int(D^1, D^3)$ are many more than in $int(D^1, D^4)$. This depends on the fact that, in the integration process we used `ScalableColor` descriptor, as external features $\overline{F}$, which correspond the descriptors used to obtain D^4 from method M^3 (criterion 2: `ScalableColor` and `DominantColor`). D^3 was obtained with method M^3 and criterion 1 (`CameraMotion` and `AudioSpectrumCentroid`). Therefore, we can conclude that the performance depends on the relationship between the external features $\overline{F}$ and the method used for the input cluster distributions (D^3 and D^4).

Finally, we observe that, during the integration process, the semantics of each cluster S_i has been characterized by the low level features ($\widetilde{F}_i$). So, at the end the process, the amount of information carried by the integrated clusterings is noticeably increased in terms of low level features as well.

5 Conclusion and future works

The proposed method represents a first approach for clustering integration. The method merges two clustering dis-

Input $\overline{F}$	$int(D^1, D^2)$	$int(D^1, D^3)$	$int(D^2, D^3)$	$int(D^1, D^4)$	$int(D^2, D^4)$
	$th = 8$	$th = 14.5$	$th = 6.5$	$th = 6$	$th = 4.5$
EH	$N_c^0 = 17$ $N_c^1 = 80$	$N_c^0 = 13$ $N_c^1 = 73$	$N_c^0 = 32$	$N_c^0 = 15$ $N_c^1 = 115$	$N_c^0 = 46$ $N_c^1 = 129$
RS	$N_c^0 = 16$ $N_c^1 = 80$	$N_c^0 = 15$ $N_c^1 = 73$	$N_c^0 = 38$	$N_c^0 = 13$ $N_c^1 = 115$	$N_c^0 = 46$ $N_c^1 = 129$
SC	$N_c^0 = 16$ $N_c^1 = 80$	$N_c^0 = 12$ $N_c^1 = 73$	$N_c^0 = 33$	$N_c^0 = 17$ $N_c^1 = 115$	$N_c^0 = 42$ $N_c^1 = 129$

Table 2: Cluster integration results (`EH=EdgeHistogram`, `RS=RegionShape`, `SC=ScalableColor`).

Integrated Distribution	**Cluster**	**Semantics**
$int(D^1, D^2)$	1	from D^1: {anchorman} from D^2: {anchor:anchor_front, anchor:anchor_front_close, anchor:anchor_angle_left}
$int(D^1, D^3)$	0	from D^1: anchorman from D^3: {open market, doctors, exhibition, fields, fishermen, church, preyers group, hippies, parliament, wine, secretary, anchorman conference, group of people, old woman, mirror, interview, house}
$int(D^1, D^4)$	1	from D^1: {anchorman} from D^4: {anchorman, boat, anchorman, woman, church, anchorman, man, woman, girl}

Table 3: Semantic cluster integration results (reference cluster distribution: D^1; external feature: `ScalableColor` descriptor).

Cluster	**Semantics**
0	jingle
1	anchorman
2	auditorium: parliament (indoor)
3	padre porta (outdoor)
4	old women (outdoor)
5	prices (indoor)
6	boats and fish (outdoor)
7	wine (outdoor-indoor)
8	auditorium: congress (indoor)
9	concert (outdoor)
10	football match (outdoor)

Table 4: Semantics of clustering method M^1.

tributions obtained with different techniques, using low level features given by further techniques (Figure 3). The integration process evaluates the granularity levels of the input clusterings, integrates the low level features in order to characterized the clusters, performs the integration at each granularity level and, finally, evaluates the cluster distribution of the output clustering.

The experimental results show that the merged cluster distribution preserves the original semantic information given by the input cluster distributions and increases the quantity of information, in terms of semantics and low level features. The results are not interesting when the quality of the input cluster distributions is low. Instead, when the external features used for the integration process are comparable with the features used by the clustering methods of input distributions, the quality reasonably increases.

Future works will investigate how to improve the weak aspects of the integration just outlined and how to improve the semantics results. It could be also worthwhile using of ontologies to better characterized the semantic terms.

This treatment considers only two granularity levels, so a possible future study consists of its generalization at n granularity levels. A further generalization can be done by considering each cluster as a general element of a given structure that has to be integrated with an other structure. The development of a general methodology could be supported by the previous work about segment decomposition [2] where each segment represents a general element of a given structure. Besides, in order to obtain a higher integration reliability, different types of low level features (color, shape, pattern, etc.) can be opportunely combined.

References

[1] http://www.research.ibm.com/videoannex/.

[2] N. Adami, M. Corvaglia, and R. Leonardi. *Comparing the quality of multiple descriptions of multimedia documents.* In *MMSP 2002*, San Thomas, Virgin Islands USA, December 2002.

[3] N. Adami and R. Leonardi. *Identification of editing effects in image sequences by statistical modelling.* In *PCS 99*, pages 157–161, Portland OR, April 1999.

[4] S. Benini, L.-Q. Xu, and R. Leonardi. *Identifying video content consistency by vector quantization.* In *WIAMIS 2005*, Montreux, CH, April 2005.

[5] M. Campanella, P. Migliorati, and R. Leonardi. *Future–Viewer: an efficient framwork for navigating and classifying audio–visual documents.* In *WIAMIS 2005*, Montreux, CH, April 2005.

[6] C. Dorai and S. Venkatesh. Bridging the semantic gap with computational media aesthetics. *IEEE Multimedia*, 10(2):15–17, April/June 2003.

[7] ISO/IEC 15938-3. *Information Technology - Multimedia Content Description Interface - Part 3: Visual*, September 2001.

[8] ISO/IEC 15938-4. *Information Technology - Multimedia Content Description Interface - Part 4: Audio*, September 2001.

[9] ISO/IEC 15938-6. *Information Technology - Multimedia Content Description Interface - Part 6: Reference Software*, September 2001.

[10] ISO/IEC JTC1/SC29/WG11 N2466. *MPEG–7 Content Set - cd 14*, Atlantic City NJ (USA), October 1998.

[11] Y.Rubner, C.Tomasi, and L.J.Guibas. *A metric for distribution with application to image database.* In *ICCV 1998*, pages 59–66, Bombay INDIA, January 1998.

Cluster	**Semantics**
0	start:na
1	jingle:na
2	anchor:studio_large
3	anchor:anchor_front, anchor:anchor_front_close, anchor:anchor_angle_right, anchor:anchor_angle_left
4	parliament:na
5	parliament:na, wine:na, congress:na
6	parliament:na
7	parliament:na, fishing:na, congress:na
8	parliament:na, pueblo:na, village:na, price_rise:na, wine:n
9	anchor:anchor_angle_left
10	pueblo:na, fishing:na
11	pueblo:na
12	pueblo:na, village:na, price_rise:na, fishing:na, wine:na
13	pueblo:na
14	pueblo:na, price_rise:na, congress:na, football1:na
15	pueblo:na, village:na, price_rise:na
16	pueblo:na
17	pueblo:na, village:na, price_rise:na, fishing:na, congress:na
18	pueblo:na, fishing:na, wine:na
19	anchor:anchor_front
20	village:na
21	village:na, price_rise:na, football1:na
22	village:na, price_rise:na, fishing:na
23	village:na, football1:na
24	anchor:anchor_angle_left
25	price_rise:na, fishing:na, wine:na, congress:na
26	price_rise:na, congress:na
27	price_rise:na
28	price_rise:na, wine:na, congress:na
29	wine:na
30	wine:na, football1:na
31	wine:na
32	anchor:anchor_angle_right
33	congress:na
34	congress:na
35	hippies_jingle:na
36	hippies_jingle:na
37	hippies_jingle:na
38	hippies_jingle:na
39	anchor: anchor_angle_right
40	football1:na
41	football1:na
42	football1:na
43	football1:na

Table 5: Semantics of clustering method M^2.

Cluster	Semantics
0	old woman, gloom auditorium
1	football match
2	world, hippies, football
3	man, car, church, fishermen, world
4	parliament, football match, prayer group
5	football match
6	woman, interview
7	football match, fishermen, party, woman
8	church, football match
9	car, football match
10	TV studio, parliament, exhibition
11	exhibition, water, crowd, stadium
12	interviews
13	man in the dark
14	parliament, hippies
15	preyers group, hippies,parliament, wine, secretary, anchorman, open market, doctors, exhibition
16	preyers group, old woman, fisherman boat, interview, conference
17	old women, fishermen,conference, man
18	world
19	window
20	conference, anchorman,group of people, church, exhibition, man, balance, fields, interview, football match
21	old woman, mirror,house, anchorman, exhibition, interview, doctor, fishermen, fields, group of people, conference
22	woman, man, interview, church, house, fishermen, fields, conference, football match, flowers

Table 6: Semantics of clustering method M^3 (non visual features).

Cluster	Semantics
0	people
1	woman
2	people
3	football
4	jingle
5	-
6	football, man
7	football, jingle
8	-
9	woman in grass
10	football
11	doctor, boat
12	football, people
13	conference
14	anchorman
15	jingle
16	boat
17	anchorman
18	anchorman, woman, church
19	man
20	countryside
21	jingle
22	boat, badge
23	countryside
24	football, bottles
25	countryside
26	men
27	anchorman, boat
28	man, girl
29	people
30	old woman
31	football
32	-
33	conference, countryside
34	-
35	-
36	man
37	people
38	jingle
39	-
40	-
41	-
42	man
43	-
44	countryside
45	countryside
46	woman
47	football
48	anchorman, man, woman, girl
49	parliament
50	man
51	football, countryside
52	jingle
53	sea, man

Table 7: Semantics of clustering method M^3 (visual features).