A semi-automatic video annotation tool with MPEG-7 content collections

Roberto Vezzani, Costantino Grana, Daniele Bulgarelli, Rita Cucchiara
DII - Università degli Studi di Modena e Reggio Emilia
{surname.name}@unimore.it

Abstract

In this work, we present a general purpose system for hierarchical structural segmentation and automatic annotation of video clips, by means of standardized low level features. We propose to automatically extract some prototypes for each class with a context based intra-class clustering. Clips are annotated following the MPEG-7 standard directives to provide easier portability. Results of automatic annotation and semi-automatic metadata creation are provided.

1. Introduction

The increasing spread of Video Digital Libraries calls for the design of efficient Video Data Management Systems to manage video access, provide summarization, similarity search, and support queries according with available annotations. Video summaries are necessary to provide compressed representations of videos without losing crucial contents and to allow efficient browsing as well as a fast overview of the original contents by dropping the time spent on tedious operations such as fast forwarding and rewind.

Examples of automatic semantic annotation systems have been presented recently, most of them in the application domain of news and sports video. Most of the proposals deal with a specific context making use of ad-hoc features. In [1] the playfield area, the number and the placement of players on the play field, and motion cues are used to distinguish soccer highlights into subclasses. Differently, a first approach trying to apply general features is described by [2]. Employing color, texture, motion, and shape, visual queries by sketches are provided, supporting automatic object based indexing and spatiotemporal queries.

We propose a general framework which allows to automatically annotating video clips by comparing their similarity to a domain specific set of prototypes. In particular, we focus on providing a flexible system directly applicable to different contexts and a standardized MPEG-7 output. To this aim, the clip characterizing features, the final video annotation, and the storage of the reference video objects and classes are realized using this standard.

Starting from a large set of manually annotated clips, according with a classification scheme, the system exploits the potential perceptual regularity and generates a set of prototypes, or visual concepts, by means of a intra-class clustering procedure. Then, only the prototypes are stored as suitable specialization concepts of the defined classes. Thanks to the massive use of the MPEG-7 standard, a remote system could then perform its own annotation of videos using these context classifiers.

2. Similarity of video clips

The problem of clip similarity can be seen as a generalization of the problem of image similarity: as for images, each clip may be described by a set of visual features, such as color, shape and motion. These are grouped in a feature vector $\mathbf{V}_i = \left[F_i^1, F_i^2, \ldots, F_i^N \right]$ where i is the frame number, N is number of features and F_i^j is the j-th feature computed at frame i. However, extracting a feature vector at each frame can lead to some problems during the similarity computation between clips, since they may have different lengths; at the same time keeping a single feature vector for the whole clip cannot be representative enough, because it does not take into account the features temporal variability. Here, a fixed number M of feature vectors is used for each clip, computed on M frames sampled at uniform intervals within the clip. In our experiments, a good tradeoff between efficacy and computational load suggests the use of $M = 5$ for clips of averaging 100 frames. To provide a general purpose system, we avoid to select context dependent features, relaying on broad range properties of the clips. To allow easier interoperability

```
<?xml version="1.0" encoding="iso-8859-1"?> <Mpeg7 [...]>
  <Description xsi:type="ClassificationSchemeDescriptionType">
    <ClassificationScheme uri="urn:mpeg:mpeg7:cs:F1_race">
      <Term termID="People" Definition=""/>
      <Term termID="Graphics" Definition=""/>
      [...]</ClassificationScheme></Description>
  <Description xsi:type="ModelDescriptionType">
    <Model xsi:type="CollectionModelType">
      <Label href="urn:mpeg:mpeg7:cs:F1_race:People"/>
      <Collection xsi:type="ContentCollectionType" id="prototype0">
        <Content xsi:type="VideoType">
          <Video><MediaLocator><MediaUri>Clip1_2323.avi</MediaUri>
          </MediaLocator></Video></Content>
        <ContentCollection><Content xsi:type="ImageType">
          <VisualFeature xsi:type="ScalableColorType" numOfCoeff="256">
            <Coeff>101376 -51 122 -100 -91694 -69 -620 -185 [...]</Coeff>
          </VisualFeature>
          <VisualFeature xsi:type="ColorLayoutType">
            <YDCCoeff>31</YDCCoeff><CbDCCoeff>20</CbDCCoeff>
            <CrDCCoeff>20</CrDCCoeff>
            <YACCoeff63>13 15 17 13 17 16 18 19 [...]</YACCoeff63>
            <CbACCoeff63>20 17 15 17 15 15 16 16 [...]</CbACCoeff63>
            <CrACCoeff63>7 15 18 15 16 20 15 16 [...]</CrACCoeff63>
          </VisualFeature>
          <GridLayoutDescriptors numOfPartX="2" numOfPartY="2">
            <Descriptor xsi:type="ParametricMotionType"
              motionModel="translational">
              <CoordDef originX="0" originY="0"/>
              <Parameters>2.828283 -0.404040</Parameters></Descriptor>
            [...]</GridLayoutDescriptors>
        </Content></ContentCollection></Collection></Model>
    [...]</Description>
  <Description xsi:type="SemanticDescriptionType">
    <Semantics id="ColorLayout">
      <Property><Name>0.40000</Name></Property>
    </Semantics>
    [...]</Description></Mpeg7>
```

Fig. 1. Example of an MPEG-7 ontology description.

and feature reuse, we tried to select features which comply with the MPEG-7 standard [4]:

1. Scalable color: a color histogram, with 16 values in H and 4 values in each S and V (256 bins in total).
2. Color layout: to account for the spatial distribution of the colors, an 8x8 grid is superimposed to the picture and the average YCbCr values are computed for each area.
3. Parametric motion: making use of the MPEG motion vectors, the translational motion of each quarter of frame is estimated.

Thus, the distance between two clips S_u and S_v is defined as

$$d(S_u, S_v) = \frac{1}{M}\sum_{i=1}^{M}\left\|\mathbf{k}^{\mathrm{T}}\left(\mathbf{V}_{u_i} - \mathbf{V}_{v_i}\right)\right\| = \frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{N} k_j \left\|F_{u_i}^{j} - F_{v_i}^{j}\right\| \tag{1}$$

where, $\mathbf{k} = (k_1, \ldots, k_N)$ is a weight vector and u_i and v_i are the frame numbers of the i-th subsampled frames of S_u and S_v respectively. The weights $k_j \in [0,1]$ provide dimensional coherence among the different features and at the same time they allow to change their relative significance.

3. Automatic annotation

Given a domain-specific video digital library, we assume that it is possible to partition the clips of videos of that context into a set of L classes $\mathbf{C} = (C_1, \ldots, C_L)$, which describe different contents or camera views. Given a large set of training clips, we assign each of them to a specific class C_k and then we employ it for automatic annotation purposes. An unknown clip can be classified using a nearest neighbor approach and the similarity measure defined above.

Since the MPEG-7 standard can naturally include pictorial elements such as objects, key-frames, clips and visual descriptors, we used it to store the classification data by means of the double description provided by the *ClassificationSchemeDescriptionType* Descriptor Schema (DS) combined with a *ModelDescriptionType* DS which includes a *Collection ModelType* DS. The classification scheme allows the definition of a taxonomy or thesaurus of terms which can be organized by means of simple term relations. The collection model is instead an *AnalyticModel,* and therefore it describes the association of labels or semantics with collections of multimedia content. The collection model contains a *Content CollectionType* DS with a set of visual elements which refer to the model being described. In particular, we linked the selected clips and a representation of their features by means of the *ScalableColor* Descriptor (D), *ColorLayout* D and the *ParametricMotion* D. In Fig. 1, an example of an MPEG-7 description of a context classifier is provided.

An advantage of adopting a MPEG-7 framework is that other systems may employ the same data enabling interoperability and easier integration.

4. Prototypes creation

As stated above, we adopt a nearest neighbor approach to classify each clip of the test sequences. Increasing the number of training clips the classification performance consequently improves, since a finer partitioning of the feature space is obtained. Unfortunately, in such a manner the space required to store the data, the corresponding transmission time -if needed-, and the computational cost for nearest neighbor selection increase.

4.1. Intra-class clustering

Since not all the clips are equally important to obtain the final classification due to perceptual redundancy in specific domains, we employ a

hierarchical clustering method, based on *Complete Link* [5], to reduce the number of clip of each class, keeping only some representative prototypes, which capture the most significant aspects of a set of clips. This technique guarantees that each clip must be similar to every other in the cluster and any other clip outside the cluster has dissimilarity greater than the maximum distance between cluster elements. For this clustering method a dissimilarity measure between two clusters W_i and W_j is defined as

$$D(W_i, W_j) = \max_{S_x \in W_i, S_y \in W_j} d\left(S_x, S_y\right). \quad (2)$$

where d is computed as in Eq. 1. The algorithm merges the most similar clusters one by one. For each class, it produces a hierarchy of clips partitions with as much levels as the cardinality of the class, where level 1 is the final step where everything is merged in a single cluster. Instead of a manual selection of the desired clustering level, or a threshold guided one, an automatic selection strategy is proposed. Let us define the cluster diameter and the cluster distance as

$$\Delta\left(W_i\right) = D\left(W_i, W_i\right), \quad (3)$$

$$\delta(W_i, W_j) = \min_{S_x \in W_i, S_y \in W_j} d\left(S_x, S_y\right) \quad (4)$$

The *Clustering Score* at level n is defined as

$$CS_n = \min\left(\Delta_1 - \Delta_n, \delta_n\right) \quad (5)$$

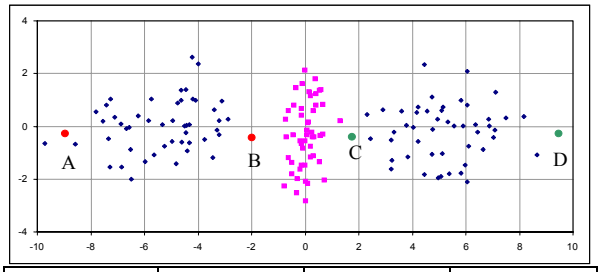
where

$$\Delta_n = \max_{W_i \in E_n} \Delta\left(W_i\right), \delta_n = \min_{W_i, W_j \in E_n, i \neq j} \delta\left(W_i, W_j\right) \quad (6)$$

The selected level is the one which minimizes CS_n. A single prototype can be generated from each cluster, by computing the average feature vectors. The clip which minimizes the distance from the prototype features is associated to it, in order to provide a representative of the visual concept. The automatic prototypes selection allows a remarkable reduction of examples. We tested in many different contexts, such as the Torino 2006 Olympic Winter Games, soccer matches, Formula 1 races, news, and the automatic clustering selects about 20% only of the provided clips as visual concepts.

4.2. Intra-class clustering with context data

The choice of significant prototypes guided by how similar the original clips are in the feature space, without considering the elements belonging to the other classes (*context data*), may lead to a prototype selection which is indeed representative of the class but lacks the properties useful for discrimination purposes. To better understand this concept, an example is reported in Fig. 2, in which 150 random



	AB	BC	CD
distance	**6.9**	**3.7**	**7.7**
dissimilarity	**849**	**2135**	**1050**

Fig. 2. Example of the effect of the context on the dissimilarity measure.

samples are extracted from three different Gaussian distributions. Two of them (blue color) belong to the same class, while the third distribution (purple color) is a different one. With the clustering technique described in the previous section, the generation of the prototypes for the first class does not take into account the second distribution; the automatic index selection could merge the two distributions leading to a single prototype. To cope with this problem, a context based similarity measure is provided as follows.

We define an *isolation coefficient* for each clip as:

$$\gamma\left(S_u\right) = \sum_{i=1, i \neq j}^{L} \sum_{S_v \in C_i} \frac{1}{d\left(S_u, S_v\right)}, S_u \in C_j. \quad (7)$$

Then we can introduce a class based *dissimilarity measure* between two clips as:

$$\bar{d}\left(S_u, S_v\right) = d\left(S_u, S_v\right) \cdot \gamma\left(S_u\right) \cdot \gamma\left(S_v\right). \quad (8)$$

The *intra-class complete link* clustering is thus enhanced with context data by substituting the clips distance in Eq. 2 and Eq. 4 with this dissimilarity measure. In Fig. 2 four samples of the blue class have been selected. Even if the central points (B,C) are closer each other than the corresponding colored ones (A and D respectively), the interposed purple distribution largely increases their dissimilarity measure, preventing their merge in a single cluster.

5. Experimental Results

The knowledge representation by means of textual and pictorial concepts is particularly suitable for summarization and fast video access. The described system has been tested on videos of different contexts: here we provide experiments to test the effectiveness of these prototypes and we propose a semiautomatic annotation framework, to speedup a metadata creation task.

We created training sets for different contexts using the first part of each video, then the rest was employed as the test set. Results are reported in Table 1. All the training clips have been reduced by the prototype creation algorithm and the number of the generated prototypes is about a third of the initial samples. Other experiments on larger training sets have shown higher reduction rates, which also depend on the number of sample per class. The results obtained on the training set show that the clustering process is able to keep the original structure of the data. On the test set, it is possible to see that the use of this approach reach a classification rate around 70% on average, depending on the context.

It is clear that without context specific features it is not feasible to reach very high automatic classification rates. We believe that 70% correct classification with a generic feature set is a realistic figure. A possible approach to the distribution of annotated videos over internet may be the use of this kind of generic tools followed by manual correction of errors, as it is common for example with every day use of OCR software. To this aim we tested on some winter Olympic Games of Torino 2006 the speedup obtained using our automatic classifier followed by a manual error correction instead of a completely manual annotation. In Fig. 3 the results over a sample video are shown. The score of the automatic classification of the non annotated clips grows with the rise of the manually annotated ones; even though, it is not convenient to annotate too much clips since the

Table 1. Classification results. NN: Nearest Neighbor using all the training clips, CL: after prototype creation with classic Complete Link clustering, CBCL: after prototype creation with Context-Based Complete Link clustering.

		Ski	**Bob**	**F1**
#Training set		300	300	500
#Test set		912	1122	1839
# of Visual Concepts	NN	300	300	500
	CL	84	126	191
	CBCL	78	122	203
Results on training set	NN	300 (100%)	300 (100%)	500 (100%)
	CL	299 (99.7%)	292 (97.3%)	478 (95.6%)
	CBCL	300 (100%)	285 (95%)	492 (98.4%)
Results on test set	NN	660 (72.4%)	854 (75.8%)	1181 (64.2%)
	CL	654 (71.7%)	846 (75.1%)	1159 (63.0%)
	CBCL	657 (72%)	852 (75.7%)	1209 (65.7%)

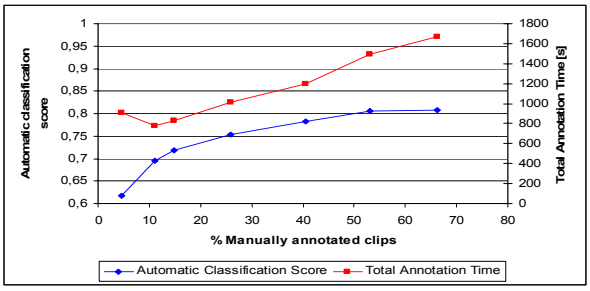


Fig. 3. Semiautomatic annotation framework example. Manual annotation of all the clips is slower than error correction of automatic tools.

increase in correct annotations does not compensate the increased time requirements. In our experiments, the best compromise is reached around 10%.

6. Conclusions

We presented a general purpose system for a hierarchical structural segmentation and automatic annotation of video clips by means of standardized low level features. A process for reducing space and computational requirements by the creation of prototypes with context based intra-class clustering was described. The classifier data are stored in MPEG-7 compliant format to improve the interoperability and integration of different systems. The annotation has shown satisfactory results without specific feature development. This approach allows a system to behave differently by simply providing a different context, thus expanding its applicability to mixed sources digital libraries. This work is supported by the DELOS NoE on Digital Libraries, as part of the IST Program of the European Commission (Contract G038-507618).

7. References

[1] M. Bertini, R. Cucchiara, A. Del Bimbo, C. Torniai, "Video Annotation with Pictorially Enriched Ontologies," in Proc. IEEE Int. Conf. Multimedia and Expo, Amsterdam, The Netherlands, 1428-1431, 2005.

[2] S. Chang, W. Chen, H.J. Meng, H. Sundaram, D. Zhong, "A Fully Automated Content-Based Video Search Engine Supporting Spatiotemporal Queries," in IEEE Transactions on Circuits and System for Video Technology, vol. 8, n. 5, 602-615, 1998.

[3] C. Grana, G. Tardini, R. Cucchiara, "MPEG-7 Compliant Shot Detection in Sport Videos," in Proc. IEEE International Symposium on Multimedia, Irvine (CA), USA, 395-402, 2005.

[4] Information technology - Multimedia content description interface - Part 3: Visual, ISO/IEC Std. 15 938-3:2003, 2003.

[5] A.K. Jain, R.C. Dubes, "Algorithms for clustering data," Prentice-Hall, Englewood Cliffs, NJ, 1988.