Retrieval of Multimedia Objects by Combining Semantic Information from Visual and Textual Descriptors*

Mats Sjöberg, Jorma Laaksonen, Matti Pöllä, and Timo Honkela

Laboratory of Computer and Information Science
Helsinki University of Technology
P.O. Box 5400, 02015 HUT, Finland
{mats, jorma, mpolla, tho}@cis.hut.fi
http://www.cis.hut.fi/picsom/

Abstract. We propose a method of content-based multimedia retrieval of objects with visual, aural and textual properties. In our method, training examples of objects belonging to a specific semantic class are associated with their low-level visual descriptors (such as MPEG-7) and textual features such as frequencies of significant keywords. A fuzzy mapping of a semantic class in the training set to a class of similar objects in the test set is created by using Self-Organizing Maps (SOMs) trained from automatically extracted low-level descriptors. We have performed several experiments with different textual features to evaluate the potential of our approach in bridging the gap from visual features to semantic concepts by the use textual presentations. Our initial results show a promising increase in retrieval performance.

1 Introduction

The amounts of multimedia content available to the public and to researchers has been growing rapidly in the last decades and is expected to increase exponentially in the years to come. This development puts a great emphasis on automated content-based retrieval methods, which retrieve and index multimedia based on its content. Such methods, however, suffer from a serious problem: the *semantic gap*, i.e. the wide gulf between the low-level features used by computer systems and the high-level concepts understood by human beings. In this paper we propose a method of using different textual features to help bridge the semantic gap from visual features to semantic concepts.

We have used our PicSOM [1] content-based information retrieval (CBIR) system with video data and semantic classes from the NIST TRECVID 2005[1] evaluation set. The TRECVID set contains TV broadcasts in different languages

* Supported by the Academy of Finland in the projects *Neural methods in information retrieval based on automatic content analysis and relevance feedback* and *Finnish Centre of Excellence in Adaptive Informatics Research.*

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

S. Kollias et al. (Eds.): ICANN 2006, Part II, LNCS 4132, pp. 75–83, 2006.

and textual data acquired by using automatic speech recognition software and machine translation where appropriate. Both the training and evaluation sets are is accompanied with verified semantic ground truth sets such as videos depicting explosions or fire.

The general idea is to take a set of example videos in the training set belonging to a given semantic class and map these onto the test set by using Self-Organizing Maps that have been trained with visual and textual feature data calculated from the video objects. This mapping generates different relevance values for the objects in the test set which can be interpreted as membership values of a fuzzy set corresponding the given semantic class. I addition to a basic set of visual and aural features, experiments comparing the retrieval accuracy with different textual features were performed. In this paper we discuss experiments using word histogram and keyword frequency features using SOMs and a binary keyword feature using an inverted file.

Section 2 describes the PicSOM CBIR system and Section 3 the TRECVID video data in more detail. Section 4 discusses how textual features can help in bridging the semantic gap between visual features and high-level concepts. The feature extraction methods are explained in Section 5 and the experiment results in Section 6. Finally, conclusions are drawn in Section 7.

2 PicSOM CBIR System

The content-based information retrieval system PicSOM [1] has been used as a framework for the research described in this paper. PicSOM uses several Self-Organizing Maps (SOMs) [2] in parallel to index and determine the similarity and relevance of database objects for retrieval. These parallel SOMs have been trained with different data sets acquired by using different feature extraction algorithms on the objects in the database. This results in each SOM arranging the objects differently, according to the corresponding feature.

Query by example (QBE) is the main interactive operating principle in PicSOM, meaning that the user provides the system a set of example objects of what he or she is looking for, taken from the existing database. This relevance information is used in the PicSOM system which expands the *relevance assessment* to related objects, such as keyframe images and textual data of a video.

For each object type (i.e. video, image, text), all relevant-marked objects in the database of that type get a positive weight inversely proportional to the total number of relevant objects of the given type. Similarly the non-relevant objects get a negative weight inversely proportional to their total number. The grand total of all weights is thus always zero for a specific type of objects. On each SOM, these values are summed into the best-matching units (BMUs) of the objects, which results in sparse value fields on the map surfaces.

After that the value fields on the maps are low-pass filtered or "blurred" to spread the relevance information between neighboring units. This produces to each map unit a *qualification value*, which is given to all objects that are mapped to that unit (i.e. have it as the BMU). Map areas with a mixed distribution of

positive and negative values will even out in the blurring, and get a low average qualification value. Conversely in an area with a high density of mostly positive values, the units will reinforce each other and spread the positive values to their neighbors. This automatically weights the maps according to relevance and coherence with the user's opinion.

The next processing stage is to combine the qualification values gained from each map to the corresponding objects. These values are again shared with related objects. The final stage is to select a specific number of objects of the desired target type with the highest qualification values. These will be returned to the user as retrieval results.

The PicSOM system has typically been used in interactive retrieval where the user can influence the response of the system with relevance feedback and the results will improve in each iteration. In this paper, however, we run only one non-interactive iteration, as we are merely interested in the mapping abilities of the SOMs for semantic concepts using textual and other features.

3 TRECVID Video Data

In 2005 our research group at Helsinki University of Technology took part in the TRECVID 2005 video retrieval evaluations [3]. The TRECVID data contains about 790 videos divided into a total of almost 100 000 video clips. From this set we picked only those that had some associated textual data and semantic classifications, resulting in a set of about 35 000 video clips. These video clips were used for the experiments described in this paper. Each video clip has one or many keyframes, which were representative still images taken from the video. Also the sound of the video was extracted as audio data. TRECVID provided textual data acquired by using automatic speech recognition software and machine translation from Chinese (Mandarin) and Arabic to English.

In the PicSOM system the videos and the parts extracted from these were arranged as hierarchical trees as shown in Fig. 1, with the main video as the parent object and the different extracted media types as child objects. In this way the relevance assessments can be transferred between related objects in the PicSOM algorithm as described in the previous section. From each media type different features were extracted, and Self-Organizing Maps were trained from these as is shown with some examples in the figure.

A large set of semantic sets were provided with the TRECVID data. These are each a set of video clips both in the training and test sets that belong to a given semantic class, for example videos depicting an exterior of a building. Table 1 shows the eight semantic classes that were used in our experiments. The first and second columns in the table give the number of videos in the training set and in the test set respectively. The given description is a shortened version of the one that was used when the classes were selected by hand during the TRECVID evaluations.

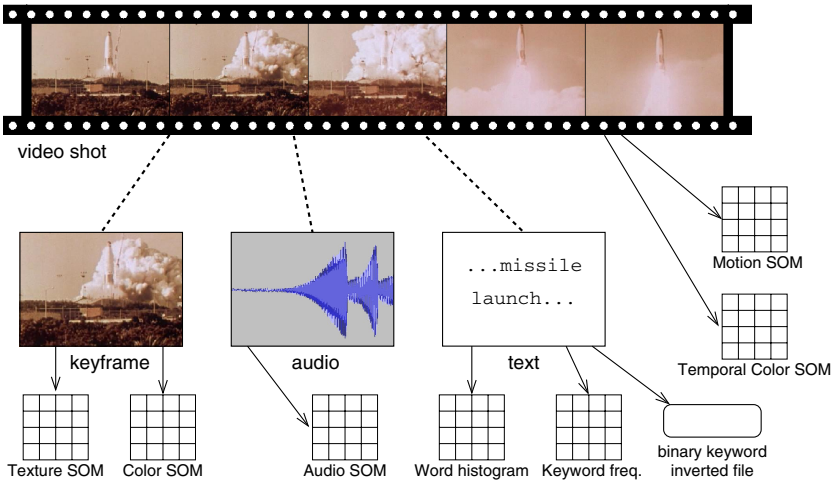

Fig. 1. The hierarchy of videos and examples of multi-modal SOMs

Table 1. Semantic classes from the TRECVID 2005 data set

training set	test set	description
109	265	an explosion or a fire
376	282	depicting regional territory graphically as a map
123	151	depicting a US flag
1578	943	an exterior of a building
375	420	depicting a waterscape or waterfront
23	32	depicting a captive person, e.g., imprisoned, behind bars
460	437	depicting any sport in action
1279	1239	a car

4 Bridging the Semantic Gap with Textual Features

The PicSOM system was initially designed for images, and particularly using visual features only. Such features describe images on a very low abstraction level, for example local color distributions, and do not generally correspond very well with the human perception of an image. In the experiments described in this paper we have also used video and aural features, but the problem remains the same: a very low-level feature description cannot match human understanding.

However, textual features do have a closer relationship to semantic concepts, as they describe the human language which has a much closer relation to the semantic concepts than for example low-level visual features. By including textual features we hope to bring the feature and concept levels closer and thus help to bridge the semantic gap. By using SOM techniques this is done in a fuzzy

manner, providing only semantic class membership values for each video, which is appropriate as such relationships can never be defined exactly, even by human beings.

Different textual features and retrieval methods exists. In this paper we will concentrate on the PicSOM system and try out three different textual features described in more detail in the following section.

5 Feature Extraction

5.1 Non-textual Features

From the videos we calculated the standard MPEG-7 Motion Activity descriptor using the MPEG-7 Experimentation Model (XM) Reference Software [4]. We also calculated our own non-standard temporal features of color and texture data.

A temporal video feature is calculated as follows. Each frame of the video clip is divided into five spatial zones: upper, lower, left, right and center. A still image feature vector is calculated separately for each zone and then concatenated to form frame-wise vectors. The video clip is temporally divided into five non-overlapping video sub-clips or slices of equal length. All the frame-wise feature vectors are then averaged within the slices to form a feature vector for each slice. The final feature vector for the entire video clip is produced by concatenating the feature vectors of the slices. For example using the 3-dimensional average RGB color still image feature we would get a final vector with a dimensionality of $3 \times 5 \times 5 = 75$. The idea is to capture how the averaged still image features change over time in the different spatial zones.

We used average RGB color, texture neighborhood and color moments each separately as a basis for the temporal feature algorithm. Texture neighbourhood is a simple textural feature that examines the luminance values of the 8-neighbourhood of each inner pixel in an image. The values of the feature vector are then the estimated probabilities that the neighbor pixel is brighter than the central pixel (given for each 8-neighborhood position).

If we treat the values in the different color channels of the HSV color space as separate probability distributions we can calculate the three first central moments: mean, variance and skewness. And when we calculate these for each of the five zones mentioned we get our third feature: color moments.

From the still images we calculated the following standardized MPEG-7 descriptors using the MPEG-7 XM software: Edge Histogram, Homogeneous Texture, Color Structure and Color Layout. Additionally we used a Canny edge detection feature which was provided by TRECVID.

From the audio data we calculated the Mel-scaled cepstral coefficient [5], i.e. the discrete cosine transform (DCT) applied to the logarithm of the mel-scaled filter bank energies. This feature is calculated using an external program created by the Speech recognition group at the Laboratory of Computer and Information Science at the Helsinki University of Technology[2].

[2] `http://www.cis.hut.fi/projects/speech/`

5.2 Word Histogram

The word histogram feature is calculated in three stages. First a histogram is calculated for each textual object (document) in the database giving the frequencies of all the words in that text. Then the document-specific histograms are combined into a single histogram or dictionary for the whole database. The final word histogram feature vectors are calculated for each document by comparing its word frequencies to the dictionary, i.e. the words in the database-wide histogram. For each word that does not belong to the list of stop words (i.e. not commonly used words such as "the") in the dictionary we calculate the tf-log-idf weight [6] for the document. The resulting feature vector then gives the tf-log-idf values for all dictionary words in that document.

The tf-idf weight is commonly used in information retrieval and is given as the product of the *term frequency* and the *inverse document frequency*. The term frequency for a word k in one document is calculated as

$$\mathrm{tf}_k = \frac{n_k}{\sum_{j \in K_D} n_j}, \tag{1}$$

where n_k is the number of occurrences of the word k and the denominator gives the number of occurrences of all dictionary words K_D in the document. The corresponding document frequency is calculated as

$$\mathrm{df}_k = \frac{N_k}{N}, \tag{2}$$

where N_k is the number of documents where the word k appears, and N is the total number of documents. The tf-log-idf is then given as the product of Eq. (1) and the log-inverse of Eq. (2):

$$\text{tf-log-idf}_k = \frac{n_k}{\sum_{j \in K_D} n_j} \log \frac{N}{N_k}. \tag{3}$$

The feature vector produced in this manner has a dimensionality of about 27 000. This is finally reduced to 100 by using singular value decomposition.

5.3 Keyword Frequency

Information about word occurrence frequencies were used to extract relevant keywords from each text document. Specifically, the frequency of occurrence for each word was compared to the corresponding frequency in another text corpus which was assumed to be neutral of domain specific terms. In these experiments, the Europarl corpus [7], extracted from European parliament proceedings, was used as the reference corpus. For each word, the ratio of the word's rank in the list of most frequent words in the document to the corresponding rank in the reference corpus was computed as an indicator of a semantically relevant keyword. Using this scheme words such as 'nuclear' would result in a high ratio despite rare occurrence in the document while words such as 'the', 'on' or 'and' would result in a low ratio regardless of frequent occurrence in the document.

5.4 Binary Keyword Features

A recent extension of the PicSOM system allows the usage of an inverted file as an index instead of the SOM [8]. The binary keyword feature is such a feature, where an inverted file contains a mapping from words to the database objects containing them.

The binary keyword features were constructed by gathering concept-dependent lists of most informative terms. Let us denote the number of video clips in the training set associated with semantic class c as N_c and assume that of these videos, $n_{c,t}$ contain the term t in the textual data. Using only non-stop words which have been stemmed using the Porter stemming algorithm [9], the following measure can be calculated for term t regarding the class c:

$$S_c(t) = \frac{n_{c,t}}{N_c} - \frac{n_{all,t}}{N_{all}} \ . \tag{4}$$

For every semantic class, we record the 10 or 100 most informative terms depending on which one gives better retrieval performance.

The inverse file is then created as mapping from these informative words to the database objects (texts) that contain them. In the PicSOM system a measure indicating the closeness of a textual object i to the semantic class c used in generating the inverse file can be calculated as

$$S_{i,c} = \sum_k \frac{\delta_{i,k}}{N_k}, \text{ where } \delta_{i,k} = \begin{cases} 1 & \text{if } k \text{ exists in } i, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

and where sum is taken over all words k in the inverse file, and where N_k is the total number of textual objects that contain the keyword k. The higher the value of $S_{i,c}$ for a specific textual object is, the closer it is deemed to be to the given class c. The value of this measure is then added to the qualification values of objects produced by the visual and aural features.

6 Experiment Results

Four experiment runs are presented in this paper, each for a different combination of features used: (i) only non-textual features, and non-textual features combined separately with (ii) word histogram, (iii) keyword frequency, and (iv) binary keyword features. The binary keyword feature used different inverted files for each semantic class as explained previously. Each experiment was performed separately for each of the eight semantic classes, and the performance was evaluated using the average precision of retrieval.

The non-interpolated average precision is formed by calculating the precision after each retrieved relevant object. The final per-class measure is obtained by averaging these precisions over the total number of relevant objects, when the precision is defined to be zero for all non-retrieved relevant objects. The per-class average precision was finally averaged over all semantic classes to generate an overall average precision.

The experiment results are summarized in Table 2, with the best results for each class indicated in bold face. The results show how the retrieval performance increases as the textual features are used. Overall the binary keyword features make a substantial improvement, while the keyword frequency and word histogram features give much smaller improvements. If we look at the class-wise results, the binary keywords feature performs best in half of the cases, often with a considerable advantage over the other methods. Keyword frequency seems to do worst overall of the three textual features, but in three cases it is still better than the others, although with a small margin only.

One explanation for the relatively bad results of the keyword frequency and word histogram features is the low quality of the textual data. Speech recognition is never perfect, and machine translation reduces the quality even more. A visual inspection of the texts shows many unintelligible words and sentences. On the other hand, a sufficient number of correct words still seem to get through to make a significant difference. The fact that the binary keyword feature compares keywords with the rest of the database instead of a task-neutral external corpus, as the keyword frequency feature does, can explain the differences as well.

Table 2. Average precision results for experiments

semantic class	non-textual	kw freq.	word hist.	binary kw
an explosion or a fire	0.0567	0.0567	0.0582	**0.0680**
map of regional territory	0.3396	0.3419	0.3418	**0.3423**
depicting a US flag	0.0713	0.0715	0.0716	**0.0808**
an exterior of a building	0.0988	**0.0993**	0.0989	0.0972
waterscape or waterfront	0.2524	**0.2525**	0.2524	0.2500
captive person	0.0054	**0.0059**	0.0058	0.0029
any sport in action	0.2240	0.2242	0.2258	**0.2675**
a car	0.2818	0.2820	**0.2843**	0.2820
overall average	0.1662	0.1667	0.1674	**0.1739**

7 Conclusions

In this paper, we have studied the mapping of semantic classes of videos from a training set to a test set using Self-Organizing Maps. The nature of the resulting presentation of a semantic class can be understood as a fuzzy set where the relevance or qualification values of the retrieved videos can be interpreted as membership values. Furthermore we have studied the effect of using textual features in addition to our original non-textual, mostly visual, features. As textual features have a closer relation to the semantic concepts as expressed in language we hope to narrow the semantic gap and as a result increase the retrieval performance of our PicSOM CBIR system.

Our initial experiments do indeed demonstrate that this arrangement improves the performance of the system somewhat, although not in all cases as much as one might have hoped for. Especially the keyword frequency feature

has some future potential as new improvements are currently being implemented. The choice of reference corpus should be pondered, for example using several corpora would decrease the dependence of a specific choice. Also using the entire TRECVID textual database set itself as a corpora should increase accuracy. Our initial results however show a great potential for this method and inspires us to continue research in this area.

References

1. Laaksonen, J., Koskela, M., Oja, E.: PicSOM—Self-organizing image retrieval with MPEG-7 content descriptions. IEEE Transactions on Neural Networks, Special Issue on Intelligent Multimedia Processing **13**(4) (2002) 841–853
2. Kohonen, T.: Self-Organizing Maps. Third edn. Volume 30 of Springer Series in Information Sciences. Springer-Verlag, Berlin (2001)
3. Koskela, M., Laaksonen, J., Sjöberg, M., Muurinen, H.: PicSOM experiments in TRECVID 2005. In: Proceedings of the TRECVID 2005 Workshop, Gaithersburg, MD, USA (2005) 262–270
4. MPEG: MPEG-7 visual part of the eXperimentation Model (version 9.0) (2001) ISO/IEC JTC1/SC29/WG11 N3914.
5. Davis, S.B., Mermelstein, P.: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. In Waibel, A., Lee, K., eds.: Readings in speech recognition. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1990) 65–74
6. Salton, G., McGill, M.J.: Introduction to Modern Information Retrieval. Computer Science Series. McGraw-Hill, New York (1983)
7. Koehn, P.: Europarl: A multilingual corpus for evaluation of machine translation. (`http://people.csail.mit.edu/~koehn/publications/europarl.ps`) Draft, Unpublished.
8. Koskela, M., Laaksonen, J., Oja, E.: Use of image subset features in image retrieval with self-organizing maps. In: Proceedings of 3rd International Conference on Image and Video Retrieval (CIVR 2004), Dublin, Ireland (2004) 508–516
9. Porter, M.: An algorithm for suffix stripping. Program **14**(3) (1980) 130–137