Video Navigation Based on Self-Organizing Maps

Thomas Bärecke[1], Ewa Kijak[1], Andreas Nürnberger[2], and Marcin Detyniecki[1]

[1] LIP6, Université Pierre et Marie Curie, Paris, France
thomas.baerecke@lip6.fr
[2] IWS, Otto-von-Guericke Universität, Magdeburg, Germany

Abstract. Content-based video navigation is an efficient method for browsing video information. A common approach is to cluster shots into groups and visualize them afterwards. In this paper, we present a prototype that follows in general this approach. The clustering ignores temporal information and is based on a growing self-organizing map algorithm. They provide some inherent visualization properties such as similar elements can be found easily in adjacent cells. We focus on studying the applicability of SOMs for video navigation support. We complement our interface with an original time bar control providing – at the same time – an integrated view of time and content based information. The aim is to supply the user with as much information as possible on one single screen, without overwhelming him.

1 Introduction

Extremely large databases with all types of multimedia documents are available today. Efficient methods to manage and access these archives are crucial, for instance quick search for similar documents or effective summarization via visualization of the underlying structure.

The prototype presented in this paper implements methods to structure and visualize video content in order to support a user in navigating within a single video. It focuses on the way video information is summarized in order to improve the browsing of its content. Currently, a common approach is to use clustering algorithms in order to automatically group similar shots and then to visualize the discovered groups in order to provide an overview of the considered video stream [1,2]. The summarization and representation of video sequences is usually keyframe-based. The keyframes can be arranged in the form of a temporal list and hierarchical browsing is then based on the clustered groups. In this paper, we use one promising unsupervised clustering approach that combines both good clustering and visualization capabilities: the self-organizing maps (SOMs)[3]. In fact, they have been successfully used for the navigation of text [4,5,6,7] and image collections [8,9,10].

The visualization capabilities of self-organizing maps provide an intuitive way of representing the distribution of data as well as the object similarities. As most clustering algorithms, SOMs operate on numerical feature vectors. Thus, video content has to be defined by numerical feature vectors that characterize it. A

H. Sundaram et al. (Eds.): CIVR 2006, LNCS 4071, pp. 340–349, 2006.

variety of significant characteristics has been defined for all types of multimedia information. From video documents, a plethora of visual, audio, and motion features is available [11,12]. We rely on basic colour histograms and ignore more sophisticated descriptors, since our primary goal of this study was to investigate the visualisation and interaction capabilities of SOMs for video structuring and navigation.

Our system is composed of feature extraction, structuring, visualization, and user interaction components. Structuring and visualization parts are based on growing SOMs that were developed in previous works and applied to other forms of interactive retrieval [7,13]. We believe that *growing* SOMs are particularly adapted to fit video data. The user interface was designed with the intention to provide intuitive content-based video browsing functionalities to the user. In the following four sections we will describe every system component and each processing step. First we present the video feature extraction. Then we will shortly describe how structuring works with growing self-organizing maps. Afterwards, a detailed description of the visualization component is given. Before concluding, the last section deals with the interaction possibilities of our system.

2 Video Feature Extraction

The video feature extraction component supplies the self-organizing map with numerical vectors and therefore they form the basis of the system. The module consists of two parts, temporal segmentation and feature extraction.

2.1 Temporal Segmentation

The video stream is automatically segmented into shots by detecting cuts. Our temporal segmentation is performed by detecting rapid changes of the difference between colour histograms of successive frames, using a single threshold. It was shown in [14] that this simple approach performs rather well. The colours are represented in the IHS space, because of its suitable perceptual properties and the independence between the three colourspace components. A simple filtering process allows the reduction of the number of false positives. The shots with an insufficient number of frames (usually less than 5), are ignored. However, the number of false positives does not have a great influence on our approach, since similar shots will be assigned to the same cluster, as discussed in the following.

2.2 Feature Extraction

In order to obtain good clustering a reasonable representation of the video segments is necessary. For each shot, one keyframe is extracted (we choose the median frame of a shot) along with its colour histograms using a specified colour space. The system supports the IHS, HSV, and RGB colour models. Apart from a global colour histogram, histograms for the top, bottom left, and right regions of the image are also extracted. The self-organizing map is trained with a vector merging all partial histogram vectors, which is then used to define each shot.

3 Structuring with Growing Self-Organizing Maps

3.1 The Self-Organizing Maps

Self-organizing maps (SOMs) [3] are artificial neural networks, well suited for clustering and visualization of high dimensional information. In fact, they map high-dimensional data into a low dimensional space (two dimensional map). The map is organized as a grid of symmetrically connected cells. During learning, similar high dimensional objects are progressively grouped together into the cells. After training, objects that are assigned to cells close to each other, in the low-dimensional space, are also close to each other in the high-dimensional space.

Our map is based on cells organized in hexagonal form, because the distances between adjacent cells are always constant on the map (see Fig. 1). In fact, in the traditional rectangular topology the distance would depend on whether the two cells are adjacent vertically (or rather horizontally) or diagonally.

The neuronal network structure of SOMs is organized in two layers (Fig. 1). The neurons in the input layer correspond to the input dimensions, here the feature vector describing the shot. The output layer (map) contains as many neurons as clusters needed. All neurons in the input layer are connected with all neurons in the output layer. The connection weights between input and output layer of neural network encode positions in the high-dimensional feature space. They are trained in an unsupervised manner. Every unit in the output layer represents a prototype, .i.e. here the center of a cluster of similar shots.

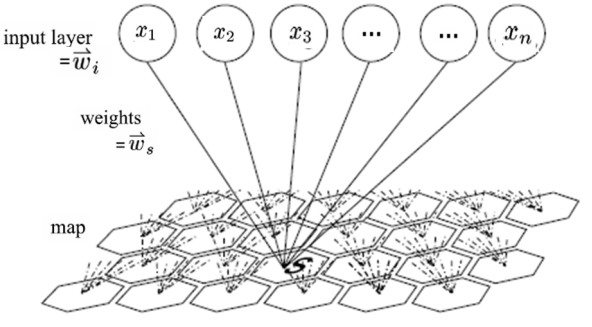


Fig. 1. Structure of a Hexagonally Organized Self-Organizing Map: The basic structure is an artificial neural network with two layers. Each element of the input layer is connected to every element of the map.

Before the learning phase of the network, the two-dimensional structure of the output units is fixed and the weights are initialized randomly. During learning, the sample vectors are repeatedly propagated through the network. The weights of the most similar prototype w_s (winner neuron) are modified such that the prototype moves towards the input vector w_i. As similarity measure usually the

Euclidean distance or scalar product is used. To preserve the neighbourhood relations, prototypes that are close to the winner neuron in the two-dimensional structure are also moved in the same direction. The strength of the modification decreases with the distance from the winner neuron. Therefore, the weights w_s of the winner neuron are modified according to the following equation:

$$\forall i : w_s^{'} = w_s + v(c,i).\delta.(w_s - w_i) \tag{1}$$

where δ is a learning rate. By this learning procedure, the structure in the high-dimensional sample data is non-linearly projected to the lower-dimensional topology.

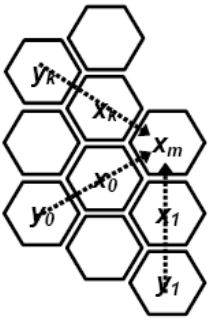
Although the application of SOMs is straightforward, a main difficulty is defining an appropriate size for the map. Indeed, the number of clusters has to be defined before starting to train the map with data. Therefore, the size of the map is usually too small or too large to map the underlying data appropriately, and the complete learning process has to be repeated several times until an appropriate size is found. Since the objective is to structure the video data, the desired size depends highly on the content. An extension of self-organizing maps that overcomes this problem is the growing self-organizing map [7].

3.2 The Growing Self-Organizing Map

The main idea is to initially start with a small map and then add during training iteratively new units, until the overall error – measured, e.g., by the inhomogeneity of objects assigned to a unit – is sufficiently small. Thus the map adapts itself to the structure of the underlying data collection. The applied method restricts the algorithm to add new units to the external units if the accumulated error of a unit exceeds a specified threshold value. This approach simplifies the growing problem (reassignment and internal-topology difficulties) and it was shown in [7] that it copes well with the introduction of data in low and high dimensional spaces. The way a new unit is inserted is illustrated in Fig. 2. After a new unit has been added to the map, the map is re-trained. Thus, all cluster centers are adjusted and the objects are reassigned to the clusters. This implies that shots may change clusters. This may lead to the emergence of empty clusters, i.e. clusters which "lost" their former objects to their neighbors. This might happen especially in areas where the object density was already small.

3.3 Similarity Between Shots

As in all clustering algorithms the main problem is how to model the similarity between the objects that are going to be grouped into one cluster. We model the difference of two video sequences with the Euclidean distance of the two vectors that were extracted from the video. However, this distance does not necessarily correspond to a perceived distance by a human. In addition, these features represent only a small part of the video content. In any case, there remains a semantic gap between the video content and what we see on the map. However, since for this first prototype study we are mainly interested in



x_i, y_i: weight vectors
x_k: weight vector of unit with highest error
m: new unit
α, β: smoothness weights
Computation of new weight vector for x_m for m:

$$x_m = \left[x_k + \alpha * (x_k - y_k) + \sum_{i=0, i \neq k}^{n} (x_i + \beta * (x_i - y_i)) \right] * \frac{1}{n+1}$$

Fig. 2. Insertion of a new Unit: When the cumulated error of a cell exceeds a threshold, a new unit x_m is added to the map. It is placed next to the unit with the highest error at the border of the map.

the capabilities of the SOMs, this approach seems sufficient, since we are not looking at grouping the shots "purely semantically", but rather at extracting a structure based on visual similarities.

4 Visualization

Our system represents a video shot by a single keyframe and constructs higher level aggregates of shots. The user has the possibility to browse the content in several ways. We combined elements providing information on three abstraction levels as illustrated in Fig. 3. First, there is an overview of the whole content provided by the self-organizing map window (see section 4.1). On each cell, the keyframe of the shot that is the nearest to the cluster centre, i.e. the most typical keyframe of a cluster, is displayed. The second level consists of a combined content-based and time-based visualization. A list of shots is provided for each grid cell (see section 4.2) and a control (see section 4.3) derived from the time-bar control helps to identify content that is similar to the currently selected shot.

4.1 Self Organizing Map Window

The self-organizing map window (see Fig. 4.1) contains the visual representation of the SOM where the clusters are represented by hexagonal nodes. The most typical keyframe of the cluster is displayed on each node. If there are no shots assigned to a special node no picture is displayed. These empty clusters emerge during the learning phase as described above. The background colours of the grid cells are used to visualize different information about the clusters. After learning, shades of green indicate the distribution of keyframes: the brightness of a cell depends on the number of shots assigend to it (see Fig. 4.1a), e.g. a cell containing four shots is displayed in a brighter green than a cell containing only two. Later, the background colour indicates the similarity of the cluster to a selected shot as described below.

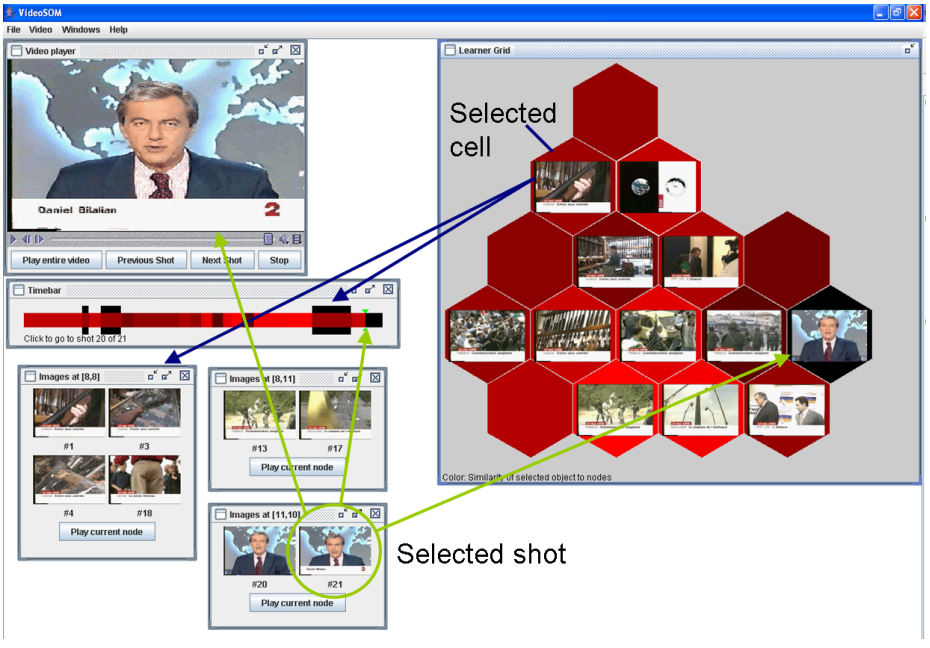
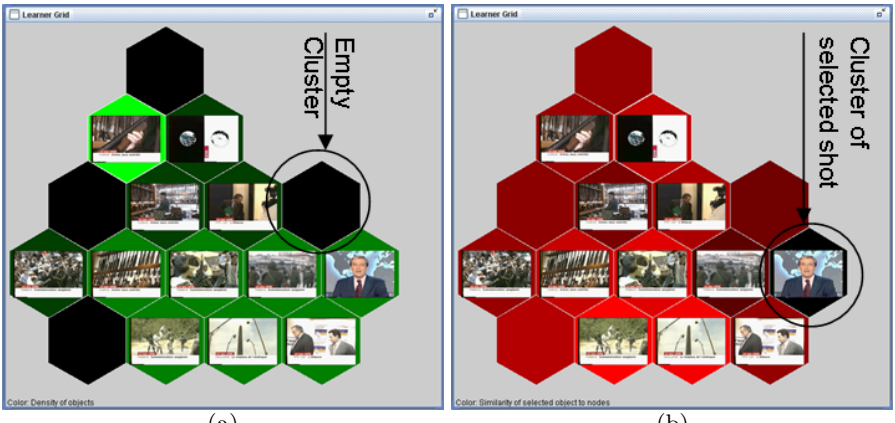


Fig. 3. Screenshot of the Interface: The player in the top left corner provides video access on the lowest interaction level. The time bar and shot list provide an intermediate level of summarized information while the growing self-organizing map on the right represents the highest abstraction level. The selected shot is played and its temporal position is indicated on the time bar whose black extensions correspond to the content of the selected cell (marked with blue arrows).

After this first display, a click on a cell opens a list of shots assigned to the specific cell (see section 4.2). The user can then select a specific shot from the list. As a result, the colour of the map changes to shades of red (see Fig. 4.1b). Here, the intensity of the colour depends on the distance between the cluster centres and the actually selected shot and thus is an indicator for its similarity. For instance, if we select a shot that has the visual characteristics A and B, all the nodes with these characteristics will be coloured in dark red and it will progressively change towards a brighter red based on the distance. This implies in particular that the current node will be automatically coloured in dark red, since by construction all of its elements are most similar. In fact, objects that are assigned to cells close to each other, in the low-dimensional space, are also close to each other in the high-dimensional space.

However, this does not mean that objects with a small distance in the high-dimensional space are necessarily assigned to cells separated by a small distance on the map. For instance, we can have on one side of the map a node with shots with the characteristic A and on another the ones with characteristic B. Then in one of both, let's say A-type, a shot with characteristics A and B.



(a) (b)

Fig. 4. Growing self-organizing map: (a) After training. The brightness of a cell indicates the number of shots assigned to each node. On each node the keyframe of the shot with the smallest difference to the cluster center is displayed. (b) After a shot has been selected. The brightness of a cell indicates the distance between each cluster center and the keyframe of the chosen shot. Notice that sequences in adjacent cells are similar as intended.

Because of the visualisation schema presented above, starting with a shot that has characteristics A and B, located in a node A, we will easily identify the nodes in which all the shots are rather of type B. This improves significantly the navigation possibilities provided by other clustering schemas.

From user interaction perspective the map is limited to the following actions: select nodes and communicate cluster assignment and colour information to the time bar. Nevertheless it is a very powerful tool which is especially useful for presenting a structured summarization of the video to the user.

4.2 Player and Shot List

The player is an essential part of every video browsing application. Since the video is segmented into shots, functionalities were added especially for the purpose of playing the previous and the next shot. A shot list window showing all keyframes assigned to a cell (Fig. 3) is added to the interface every time a user selects a node from the map. Multiple shot lists for different nodes can be open at the same time. They correspond to the actual selected node in the self-organizing map, as described in section 4.1. When clicking on one of the keyframes, the system plays the corresponding shot in the video. The button for playing the current node starts a consecutive play operation of all shots corresponding to the selected node, starting with the first shot. This adds another temporal visualization method to the segmented video.

4.3 Time Bar

The time bar (Fig. 5) provides additional information and some special interaction possibilities. A green double arrow displays the current temporal position within the video. The main bar is a projection of the colours of the self organizing map into the temporal axis. With this approach, it is possible to see within the same view the information about the similarity of keyframes and the corresponding temporal information. Additionally, corresponding shots of the selected cell are marked by black extensions. This cell can differ from the cluster of the currently selected shot, in which case the black bars correspond to the selected cluster while the colour scheme is based on the selected shot from another cluster. Thus, leading to a possibility to compare a family of similar shots with a cluster. There are two interactions possible with the time bar. By clicking once on any position, the system plays the corresponding shot. Clicking twice, it forces the self organizing map to change the currently selected node to the one corresponding to the chosen frame. And therefore, the background colour schema of the map is recomputed.

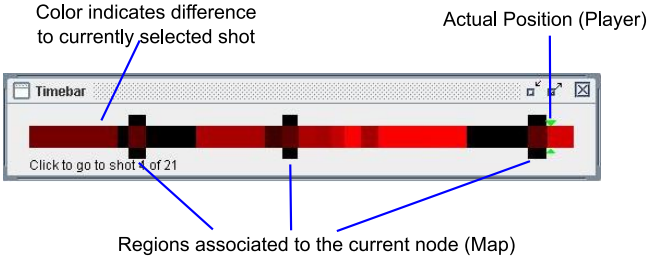


Fig. 5. Time Bar Control: The time bar control provides additional information. The brightness of the colour indicates the distribution of similar sequences on the time scale. Around the time bar, black blocks visualize the temporal positions of the shots assigned to the currently selected node. Furthermore, the two arrows point out the actual player position.

5 User Interaction

The four components presented above are integrated into one single screen (Fig. 3) providing a structured view of the video content. The methods for user interaction are hierarchically organized (Fig. 6). The first layer is represented by the video viewer. The shot lists and timebar visualize the data on the second layer. The self-organizing map provides the highest abstraction level.

The user can select nodes from the SOM and retrieve their content i.e. the list of corresponding keyframes. The time bar is automatically updated by visualizing the temporal distribution of the corresponding shots when the current node is changed. Thus, a direct link from the third to the second layer is established. Furthermore the user views at the same time the temporal distribution

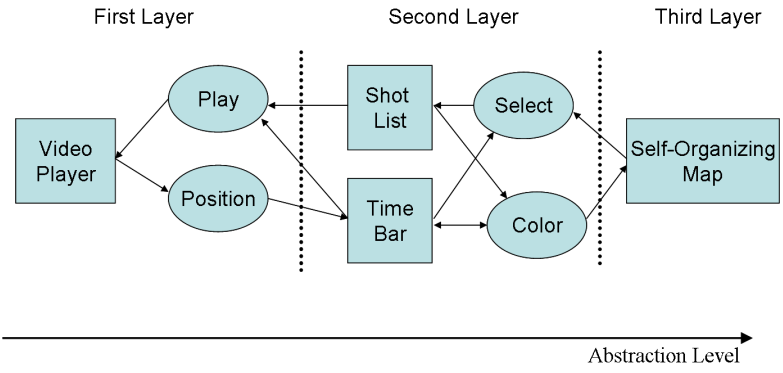


Fig. 6. User Interactions: This figure illustrates the main user interactions that are possible with our system. All listed elements are visible to the user on one single screen and always accessible thus providing a summarization on all layers at the same time.

of similar shots inside the whole video on the time bar, after a certain shot has been selected. In the other direction selecting shots using both the time bar and the list of keyframes causes the map to recompute the similarity values for its nodes and to change the selected node. The colour of the grid cells is computed based on the distance of its prototype to the selected shot. The same colours are used inside the time bar. Once the user has found a shot of interest, he can easily browse similar shots using the colour indication on the time bar or map. Notice that the first layer cannot be accessed directly from the third layer.

Different play operations are activated by the time bar and shot lists. The player itself gives feedback about its current position to the time bar. The time bar is actualized usually when the current shot changes. All visualization components are highly interconnected. In contrast to other multi-layer interfaces, the user can always use all provided layers simultaneously within the same view. He can select nodes from the map, keyframes from the list or from the time bar, or even nodes from the time bar by double-clicking.

6 Conclusions

The structuring and visualization of video information is a complex and challenging task. In this paper we presented a prototype for content-based video navigation based on a growing self-organizing map where the interaction model is hierarchically organized. We perform the clustering ignoring the temporal aspect of video information and reintroduce it in the form of a time bar where we link similar shots with colours. Our interface allows the user to browse the video content using simultaneously several perspectives, temporal as well as content-based representations of the video. Combined with the interaction possibilities between them this allows efficient searching of relevant information in video content.

References

1. Lee, H., Smeaton, A.F., Berrut, C., Murphy, N., Marlow, S., O'Connor, N.E.: Implementation and analysis of several keyframe-based browsing interfaces to digital video. In Borbinha, J., Baker, T., eds.: LNCS. Volume 1923. (2000) 206–218
2. Girgensohn, A., Boreczky, J., Wilcox, L.: Keyframe-based user interfaces for digital video. Computer **34**(9) (2001) 61–67
3. Kohonen, T.: Self-Organizing Maps. Springer-Verlag, Berlin Heidelberg (1995)
4. Lin, X., Marchionini, G., Soergel, D.: A selforganizing semantic map for information retrieval. In: Proc. of the 14th Int. ACM/SIGIR Conference on Research and Development in Information Retrieval, New York, ACM Press (1991) 262–269
5. Kohonen, T., Kaski, S., Lagus, K., Salojärvi, J., Honkela, J., Paattero, V., Saarela, A.: Self organization of a massive document collection. IEEE Transactions on Neural Networks **11**(3) (2000) 574–585
6. Roussinov, D.G., Chen, H.: Information navigation on the web by clustering and summarizing query results. Information Processing & Management **37**(6) (2001) 789–816
7. Nürnberger, A., Detyniecki, M.: Visualizing changes in data collections using growing self-organizing maps. In: Proc. of Int. Joint Conference on Neural Networks (IJCANN 2002), IEEE (2002) 1912–1917
8. Laaksonen, J., Koskela, M., Oja, E.: PicSOM-self-organizing image retrieval with MPEG-7 content descriptors. IEEE Transactions on Neural Network **13** (2002) 841–853
9. Koskela, M., Laaksonen, J.: Semantic annotation of image groups with self-organizing maps. In Leow, W.K., Lew, M.S., Chua, T.S., Ma, W.Y., Chaisorn, L., Bakker, E.M., eds.: Proc. of the 4th Int. Conf. on Image and Video Retrieval (CIVR 2005). Volume 3568 of Lecture Notes in Computer Science., Berlin, Springer-Verlag (2005) 518–527
10. Nürnberger, A., Klose, A.: Improving clustering and visualization of multimedia data using interactive user feedback. In: Proc. of the 9th Int. Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems. (2002) 993–999
11. Marques, O., Furht, B.: Content-based Image and Video Retrieval. Kluwer Academic Publishers, Norwell, Massachusetts (2002)
12. Veltkamp, R., Burkhardt, H., Kriegel, H.P.: State-Of-The-Art in Content-Based Image and Video Retrieval. Kluwer (2001)
13. Nürnberger, A., Detyniecki, M.: Adaptive multimedia retrieval: From data to user interaction. In Strackeljan, J., Leivisk, K., Gabrys, B., eds.: Do smart adaptive systems exist - Best practice for selection and combination of intelligent methods. Springer-Verlag, Berlin (2005)
14. Browne, P., Smeaton, A.F., Murphy, N., O'Connor, N., Marlow, S., Berrut, C.: Evaluating and combining digital video shot boundary detection algorithms. In: Proc. Irish Machine Vision and Image Processing Conference, Dublin (2000)