GRAPH-BASED VISUAL INTERFACES FOR NAVIGATION IN INDEXED VIDEO CONTENT[1]

Maylis DELEST, Anthony DON, Jenny BENOIS-PINEAU

LaBRI UMR CNRS 58000/University Bordeaux 1
{jenny.benois,maylis,don}@labri.fr

ABSTRACT

Indexing and segmenting video by motion, color and time has been intensively explored leading to a usual representation in a story-board. In this paper, after indexing by a spatio-temporal color signature, we describe a method allowing to navigate and to cluster the shots using an underlying graph of shots.

1. INTRODUCTION

Structuring video documents into logical video segments such as shots, scenes, objects and indexing them by characterizing their motion, color or texture is a key feature of the MPEG7 [8] standard. Numerous descriptors can be used and several techniques have already been proposed to describe video content by means of low-level features. Then, queries can be realized on indexed video in order to re-build or explore it.

For this purpose, story-boards or linear navigation interfaces[1''] are widely used to visualize the content of video documents, but they do not show the global structure of a video document nor allow an easy identification of similar shots that are distant in time. Based on structure of the document, the navigation interface has to ensure intuitive and attractive tools for browsing through video. It is natural to expect such tools will give to the user capability to access specific segments of video content and also show the structural dependencies of these segments (shots) across the document.

After video segments have been characterized by low-level descriptors, an aggregation of them into higher order structural content units, often named "scenes" becomes possible.

In this paper, we limit ourselves to a particular descriptor, namely spatio-temporal color signature (SCS) proposed in [5] and focus on the opportunity of graph-based visual interfaces for clustering shots into scenes.

Graph-based interfaces represent today a very active research field. The main reason is surely in the fact that from all data structures one can easily extract a graph representation. This fact is in conjunction with the fact that it has been proved by our visual perception is very efficient and that when algorithm cannot connect objects, the eyes can if a good representation is given to the user. Our approach is clearly in the field of information visualization and more precisely in the Schneiderman's paradigm [10]: "Overview first, zoom and filter, then details on demand". First attempts to use graph theory formalisms for video structuring and navigation were done in pioneer work [12]. Nevertheless, due to their planar structure the proposed story graphs has not found their route to users. In this paper we try to go further and propose to use huge graph navigation techniques for structuring and navigation in video documents.

In a video document, where each shot is characterized by a vector or scalar descriptor, a distance between shot descriptors can be computed.

Based on SCS-distance matrix for shots, we consider a video document as a valuated complete graph for which the distance matrix is in fact an incidence matrix. The vertices of the graph are the shots and the edges between two vertices are valuated by the corresponding entry in the SCS-distance matrix.

The main problem is now to find a method for selecting a subset of relevant distances (edges) for scene grouping. This is done by statistical methods. Then, Strahler numbers [3] on graphs can be used to build clusters of similar shots according to Strahler values of vertices.

We also experiment a 3D navigation in the clustered graph using Tulip, a software tool for visualization of large graphs. The first results given here show that this kind of exploration for video can be a nice alternative to story boards.

This paper is organised as follows. Section 2 describes the spatio-temporal similarity measure between

[1] This work was supported by the Conseil Régional d'Aquitaine and the STIC Department of the CNRS.

shots based on discrete projection transform and vector quantization. In section 3, we present Tulip, a software tool that allows visualisation and manipulation of huge graphs. Section 4 explains how a distance-matrix can be processed to obtain a valuated oriented graph. Sections 5 and 6 describe application of structural graph clustering methods to grouping shots in a video document. First results are discussed and perspectives of this work are outlined in the Section 7.

2. VIDEO STRUCTURE INDEXING

2.1. Video document model

A video document can be decomposed into a hierarchy of scenes and shots. A shot represents a continuous camera work. Shots with same semantics of their content could be grouped into scenes. We will not here discuss the complex problem of scene definition, which strongly depends on the type of video document. We limit ourselves to the assumption, that the visual semantics could be sufficiently well represented by low-level descriptors. Thus, a specific distribution of color in the image plan surely allows the separation of interior and exterior views, the properly measured intensity of motion activity allows to distinguish between strong action and nearly static narrative content. The examples here are numerous.

In the rest of this paper we suppose that the initial segmentation of a video document in the set of consecutive shots has been done. For this purpose we can use one of the methods we previously developed [5] based on a kind of discrete Radon transform called Mojette [9].

2.2. Scene grouping

Scene grouping consists of re-building of semantics of a video document from low level temporal segmentation and description : shots and their descriptors. The main idea here is to consider that temporarily adjacent shots with similar descriptor values belong to the same scene. Colour and texture descriptors such as those described in MPEG7, Part 3, could be candidate descriptors.

Therefore a distance measure between shots based on both temporal and descriptor-based distance must be set-up. In this paper, we use the results of our previous work and defining the spatio-temporal color signature of shots [5] and use it as shots' descriptor.

2.3. Weighted spatio-temporal similarity measure

Spatio-temporal colour signature of a video shot is a specific descriptor which enables measuring similarity between shots. The signature we proposed in [5] is based on a 1D discrete projective trasform called "Mojette". Mojette transform [9] is a discrete version of Radon transform used in tomography. It is done by projecting an image $(f(k,l))_{1\le k\le k',\,1\le l\le l'}$ along straight lines with different angles θ such that $\tan(\theta)=q/p$ with p and q being mutually prime integers.

Each projection indexed by p and q is a variable-size vector with each element, or bin, being the sum of pixel values along the direction given by θ. The value of a Mojette vector element is :

$$x_{p,q}[f](m)=\sum_{k=1}^{k'}\sum_{l=1}^{l'} f(k,l)\delta(m+kq-lp)$$

where δ is a Kronecker symbol and m is an integer representing the Mojette index. In the case of projections along the Ox and Oy axes with (p=1,q=0) and (p=0,q=1) respectively, the Mojette projection represents a simple summation of video frames along lines and columns. When normalized projection is computed, with the value of bins normalized by the number of pixels participating in the summation above, then the spatial distribution of the color in the direction of the axis orthogonal to the projection vector is preserved. This is an interesting property of this transform compared to usual histograms.

When all the images of a shot are transformed into 1D signals using one direction of projection, then the global color descriptor of this set of projections have to be defined to characterize the shot.

A coding–based approach was proposed in [1] to measure a similarity between shots, based on vector quantization using image blocks of all images in a shot as vectors. A vector quantization dictionary $Dico_i$ can be associated to each shot S_i. In our case quantized vectors are fixed-size vectors of Mojette transforms of images associated with each frame of a shot. We call spatio-temporal color signature of a video shot S_i its dictionary $Dico_i$ in color Mojette transform domain. Each dictionary is computed using a SplitLBG [7] method which is a generalized K-means clustering.

The use of Mojette transform in this context is very interesting because it keeps spatial distribution of color in image while reducing the computational complexity. A block-vector of dimension n^2 with a projection direction (p,q) will lead to a projection vector of dimension $(n-1)(|p|+|q|)$ as it can be derived from [9].

Using Mojette transform and vector quantization, a single color projection vector can be computed in YUV color space by the concatenation of all projected color components.

Let $I_{j,k}$ be the kth frame of the shot j, let $x_{p,q}[I_{j,k}]$ be a vector of Mojette projection with (p,q) direction. Let $x_{p,q}[I_{j,k}](m)$ be the Mojette transform value at bin m

$V(x_{p,q}[I_{j,k}])$ be the vector in the dictionary $Dico_i$ that codes vector $x_{p,q}[I_{j,k}]$:

$$d(x_{p,q}[I_{j,k}],Dico_i)=\sum_{m=1}^{\dim(x)}(x_{p,q}[I_{j,k}](m)-V(x_{p,q}[I_{j,k}])(m))^2$$

is the coding distortion when one uses a dictionary designed for the shot S_i to represent a vector of frame of the shot S_j.

Based on coding distorsion, a similarity measure $e(S_i,S_j)$ between shots S_i and S_j can be computed as proposed in [1] :

$$e(S_i,S_j)=|D_j(S_i)-D_i(S_i)|+|D_i(S_j)-D_j(S_j)|$$

where

$$D_i(S_j)=\sum_{k=1}^{NB_j} d(B_{j,k},Dico_i)$$

and NB_j is the number of blocks within the shot S_j.

The final weighted spatio-temporal similarity measure is obtained by taking the temporal distance [12] between shots into account :

$$d(S_i,S_j)=d_{temp}(S_i,S_j)\ e(S_i,S_j)$$

with

$$d_{temp}(S_i,S_j)=min(|n_b(S_i)-n_e(s_j)|,|\ n_b(S_j)-n_e(s_i)|)$$

where $n_b(S_i)$ is the number of the first frame of S_i and $n_e(S_i)$ is the number of the last frame of S_i.

2.4. Results

This technique [5] was performed on a MPEG-1 video sequence called "La Joueuse de Tympanon", SFRS®. There are 40501 frames in CIF format grouped into 200 shots. The color space used is YUV. In order to reduce computational complexity, we used DC image for each of the three color space components giving a resolution of 44x30 pixels, each pixel representing the mean value of a 8x8 pixels block in the original frame.

Two normalized Mojette vectors were computed on each frame with $p=0, q=1$ (resp. $p=1, q=0$). Each bin of the vector corresponding to the sum of pixel values along the column (resp. the line associated to the bin). Thus, we get a distance matrix

$$(d(S_i,S_j))_{1\le i\le 199,i+1\le j\le 200}.$$

It is clear that such a matrix can be considered as the incidence matrix of a valuated complete graph. In the following, we describe methods in order to cluster and navigate in this graph.

3. NAVIGATING IN HUGE GRAPHS

The main problem in handling huge graphs is the number of elements (edges and vertices) but also the management of the whole data structure (cost of time and memory). Some operations such as displaying labels, displaying clusters or managing colors can lead to an unvaluated solution.

In this work, we use the Tulip software[2]. The main features relevant in this software are:

- a powerful kernel,
- extensibility by plugins without recompiling,
- possibility to map texture and colors on edges and nodes without loosing "performance",
- easy management of cluster of clusters.

The figure 1 shows a shot of the main screen of Tulip.

A direct effect of application of the visualization techniques to the complete graph representing a video document is the possibility to "look-and-feel" the general esthetics of a movie. Thus, associating the most representative color of Mojette projections of frames in a shot to the corresponding node in the complete graph and proposing a simple linear color interpolation for coloring of edges based on the distance between nodes, we can get an idea of the whole color of the graph. Figure 5 shows the complete graph for the documentary "La Joueuse de Tympanon" ® SFRS mapped on a disk. Here the shots-nodes are situated on the borders and edges-inside. It can be felt that the dominant color of the indexed movie is brown with some light spots of white and blue.

If the grouping of shots is supposed to be performed with graph clustering methods, then the minimal weighted spanning trees can be used. A fruitful strategy here would consist in a pre-filtering of the graph in order to suppress useless edges and also orienting some edges in order to get more information on the distance-based relation of nodes.

One of our main reasons to do this is that when a graph is an oriented one, several algorithms, implemented in Tulip, are proved to be efficient for clustering or drawing.

[2] Download at www.labri.fr/~auber/projects/tulip

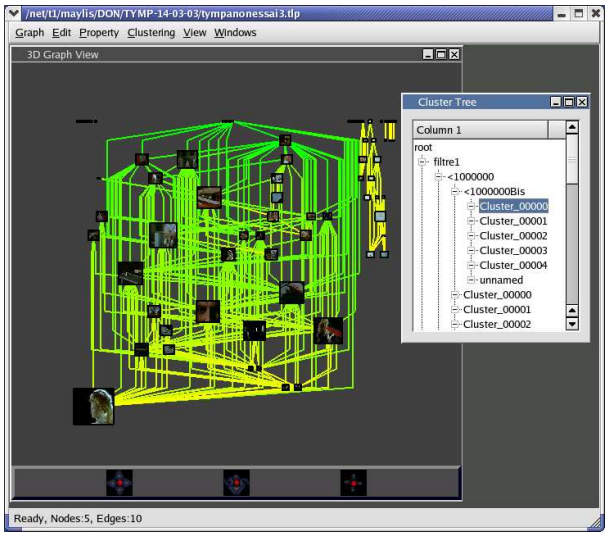

Figure 1. A view of Tulip software.

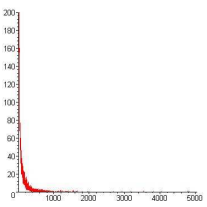
Figure 2. Histogram of $d(S_i,S_j)$.

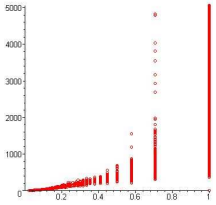
Figure 3. Function $\overline{\delta}:\delta(x)\rightarrow x$

4. FROM DISTANCE MATRIX TO VALUATED ORIENTED GRAPHS

In this section we will describe the filtering of the complete graph consisting in suppressing "weak" edges. In order to explore the distance function defined on the set of graph edges, in figure 3, we have plotted the histogram of the values $d(S_i,S_j)$ using a step of *50 000*. Let $h(x)$ be this function. Taking into account the shape of the histogram we suggest an application of the transformation

$$f(x)=\sqrt{\frac{1}{1+x}}\ .$$

Let $\delta(x)$ be the function $f(h(x))$. Then, in Figure 3, we get a representation of δ where x is the ordinate and $\delta(x)$ is the abscise. We call this application $\overline{\delta}$.
Using this application we will now filter and orient the initial complete graph.

First, we apply a linear adjustment of $\overline{\delta}$, which leads to the equation

$$\overline{\delta}_T(z)=A_0 z-B_0.$$

Of course, the linear correlation coefficient is highly significant. Thus, the variable $\varepsilon(z)=\overline{\delta_T}(z)-\delta(z)$ follows a Gaussian law. Let e be an edge having a distance value η such that $\delta(\eta)=z_\eta$ and $\overline{\delta_T}(z_\eta)-\overline{\delta}(z_\eta)$ is outside the confidence interval of $\varepsilon(z)$ computed on the interval $[0..z_n]$, then, we suppress the edge e in the graph.

After that, the same computation on the remaining edges is repeated. This leads to the equation

$$\overline{\delta}_T(z)=A_1 z-B_1$$

In our experiments on the movie "La Joueuse de Tympanon" we got values of

$A_0=841.82, B_0=51.38, A_1=396.11, B_1=7.82$

With the same computation as in the previous step, we mark the edges false. Remaining edges are marked true.

We construct now an oriented graph in the following way. Consider that all the edges (S_i,S_j) at the beginning are oriented from S_i to S_j. Let $e=(u,w)$ be an edge then if e is marked false then suppress e and add (w,u). The underlying idea is that if the distance is small then the edge will be an edge of the spanning tree. Thus all the edges market as true will be those of a spanning tree.

5. APPLICATION OF STRAHLER CHARACTERISATION TO FILTERED SHOT GRAPH

In this paper, we have only studied the effect of the Strahler parameter [2] that consists in a valuation of vertices guided by the distance on edges. This valuation takes into account the ramifications and cycles of a graph.

The Strahler number σ_b has first been introduced on binary trees in some work about the morphological structure of river [11]. It consists in associating an integer value to each node of a binary tree. These values give quantitative information about the complexity of each sub-tree of the original tree. The algorithm is a recursive one. Let s be a vertex of a tree then

- If s is a leave then $\sigma_b(s)=1$,
- Else s has two sub-trees of root s_1 and s_2 and

$$\sigma_b(s)=\begin{cases}\max(\sigma_b(s1),\sigma_b(s2)) \text{ if } \sigma_b(s1)\neq\sigma_b(s2)\\ \sigma_b(s1)+1 \text{ otherwise}\end{cases}$$

A generalization to tree can be done using the nice interpretation by Ershov [6] who has proved that the Strahler number of the root of the binary tree incremented by one is exactly the minimal number of registers needed to compute an arithmetical expression given by the tree output by the syntactical analysis. Thus, one can define the Strahler numbers σ on trees by:

$$\sigma(s)=\begin{cases}1 \text{ if } s \text{ is a leave}\\ \\ \underset{0\le i\le k}{Max}(\sigma(s_i)+i) \text{ if } s \text{ has } k+1 \text{ subtrees with root } s_i\end{cases}$$

The extension to graph [2] reuses the idea of Ershov. The generalization to directed acyclic graphs (DAG) needs no modification of the algorithm. In fact, if we consider that no result are shared in the evaluation of an expression, we can transform a DAG in a forest and then compute the Strahler number on each tree of the forest. To build a similar parameter on graphs, we consider that any graph can be viewed as a sequential program.

In a program:

- Registers are used to compute expressions. In our case, a spanning DAG enables to view a graph as an expression.
- Stacks are used to manage recursive calls. In our case, cycles in the graph are interpreted as recursive calls in the program.

Thus, we define a two dimensional parameter on rooted maps (i.e. in our case valued oriented graph). The first dimension σ represents the number of registers needed to evaluate the program, the second dimension ρ gives the number of necessary stacks (or nested call to the stack).

Here, we have built an oriented graph (section 4), and a natural spanning tree T has been built too. The valuation σ is given by the Strahler number computed on this spanning tree. Computing the parameter ρ needs to split the set of edges E in four sets which are classical when one wants to deal with connected compounds in graph :

- Tree edges (E_T) such that $e \in T$,
- Descent edges (E_D) such that $e=(u,v)$, $e \notin T$, there exists a path from u to v in T,
- Return edges (E_R) such that $e=(u,v)$, $e \notin T$, there exists a path from v to u in T,
- Cross edges (E_C) such that $e \notin T \cup E_D \cup E_R$.

Then, for each node we compute the number of used and free stacks. The algorithm is given in Figure 4. Even if the computation is a bit complicate, it has been proved [2] that an upper bound of the time complexity is $O(mlog(m))$ with m the number of edges.

By definition, in this paper, for a vertex s with valuation $\sigma(s)$ and $\rho(s)$, the Strahler number will be $\sqrt{\sigma(s)^2+\rho(s)^2}$.

```
FreeUsedToFree(node η)
      begin
      Let L a list of pair of integers (i,j)
      For all out going edges λι of η do begin
       ηi=target(λi);
       case λi in
          tree-edge set :
             ToFree[η]=0
             FreeUsedToFree (ηi)
             L.insert([Free[ηi]+ToFree[η],
                       Used[ηi]-ToFree[η]])
          cross-edge set :
             L.insert([Free[ηi]+Used[ηi],0])
          return-edge set:
             ToFree[ηi]=ToFree[ηi]+1
             L.insert([0,1])
       end case\\
      end for\\
      Used[η]=Free[η]=0
      Sort L in the decreasing order induces
           by the first dimension.
      For all element Ci=[ai,bi] in L do begin
          Used[η]=Used[η]+bi
          Free[η]=max(Free[η],ai+bi)-bi
      end for
      Return Free[$η]+Used[η]
end FreeUsedToFree
```

Figure 4. Algorithm for computing stacks number [4].

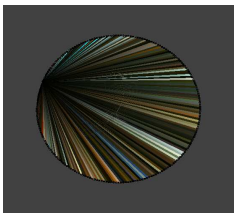

Figure 5. Distance Matrix view.

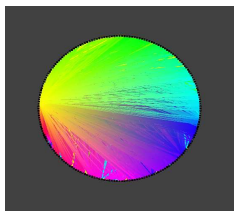

Figure 6. Distance Matrix Strahler view

First, we apply it on the distance matrix graph coloring the vertex according to the Strahler valuation from pink (high value) to yellow (low value). The effect is displayed in Figure 6. Knowing that shots-nodes of the graph are drawn on a circle in the trigonometric order and that the pink east point represents the first shot of the movie, one can see that the color reflects the ordering of shots.

6. APPLICATION OF GRAPH-CLUSTERING METHODS.

After pre-filtering of the complete graph and orientating its edges we have used a clustering method [3] based on convolution of Shtraler metrics on the oriented graphs. Thus the original distance matrix implied an oriented graph for which the Strahler metrics was associated to nodes. Then clustering nodes with associated

similar Strahler metrics allow for taking into account the induced structure of the graph. The clustering method allows for graph splitting based on a distance threshold or on a target number of clusters. Note that this method could be applied recursively to the resulting sub-graphs. We refer the reader to [3] for the complete description of the method. We chose to split the graph into five clusters. Figure 7 gives a view of the five clusters of the movie "La Joueuse de Tympanon" SFRS® in which one can navigate, using the mouse, seeing the image abstracting a shot or clicking the node in order to see the video sequence. Figures 8 and 9 show the navigation in a cluster. Figure 10 illustrates the performance of the clustering method and the underlined Strahler parameter: the majority of "graphic shots" of the movie were grouped together in one cluster. This can be explained both by the nature of the initial distance (as coding of projected images) but also by structuring capacity of the approach with regard to the initial distance matrix. Several edges in Figure 9 between thumbnail key-frames displayed show strong links between corresponding nodes, thus reflecting the similarity of shots even if they are not consequent in the movie (shots numbers are displayed in the white boxes in this figure). One can note that shots are in the same cluster because they are in the same color or they are close in time. Tulip draws automatically the sequences close one to each other because the underlying guide of the layout is a DAG.

7. RESULTS AND FURHTER PERSPECTIVES

This paper describes our first attempt to apply huge graph visualisation methods to structuring and navigation in video content. Despite the fact that only one descriptor has been used in our experiments, and only one similarity measure (distance) based on coding distortion has been studied, the results obtained on a real-world movie "La Joueuse de Tympanon" are promising. The user gets an intuitive tool for navigating inside a video document.

In the future, we plan to explore the use of these methods and the underlying Tulip software tool with a richer set of low-level descriptors of video content, such as motion and object based descriptors. In this case different metrics should be studied and experiments on user satisfaction have to be conducted.

Another open question remains on parameterisation of graph-clustering algorithms in supervised and unsupervised manner when applied to different types of video content. This is the subject of our actual work.

8. REFERENCES

[1] N. Adami, A. Bugatti, R. Leonardi, P. Migliorati, L. Rossi, Describing multimedia documents in natural and semantic-driven ordered hierarchies, Proc. ICCASP'2000, Istambul, Turkey, June 2000, IV:2023-2026.

[2] D. Auber, Using Strahler numbers for real time visual exploration of huge graphs, International Conference on Computer Vision and Graphics, 56-69, Zakopane, Pland, 25-28 September, 2002.

[3] D. Auber, Y. Chiricota, M. Delest, Graph and software clustering from valuations, LaBRI 2003, submitted to IEE Int. Conf on Software Engineering and formal methods'2003

[4] Barbieri, M., Mekenkamp, G., Ceccarelli, M.P., Nesvadba, J. « The color browser: a content driven linear :video browsing tool, ICME 2001 (Int. Conf. on Multimedia and Expo), , Tokyo, Japan, August 22-25 2001.

[5] J. Benois-Pineau, W. Dupuy, D. Barba, Re-covering of visual scenarios in movies by motion analysis and grouping spatio-temporal colors signatures of video shots, Eusflat 2001 (Internationnal Conference on Fuzzy Logic and Technology),Leicester,Septembre,2001.

[6] A. P. Ershov, On programming of arithmetic operations, Com, of the A.C.M, 1958, 8, 3-6.

[7] R.M. Gray, Vector quantization, IEEE ASSP Magazine, 1984, Avril, 4-28.

[8] ISO/IEC JTC 1/SC 29/WG 11/M6156, MPEG-7 Multimedia Description Schemes WD (Version 3.1), Beijing, July 2000.

[9] N. Norman, J.P. Guedon, La transformée Mojette : une representation redondante pour l'image, Compte Rendu à l'Académie de Sciences, Paris, 1998, 326(1), 123-126.

[10] B. Shneiderman, The eyes have it: A Task by Data Type Taxonomy for Information Visualization, IEEE Conference on visual languages, Boulder, 1996, 336-343.

[11]A. N. Strahler, Hypsomic analysis of erosional topography, Bulletin Geological Society of America 1952, 63, 1117-1142.

[12] M. Yeng, B-L Yeo, "Time-constrained Clustering fro Segmentation of Video into Story Units", Proc; of ICPR'96, vol.3, pp.375 – 380, August 1996

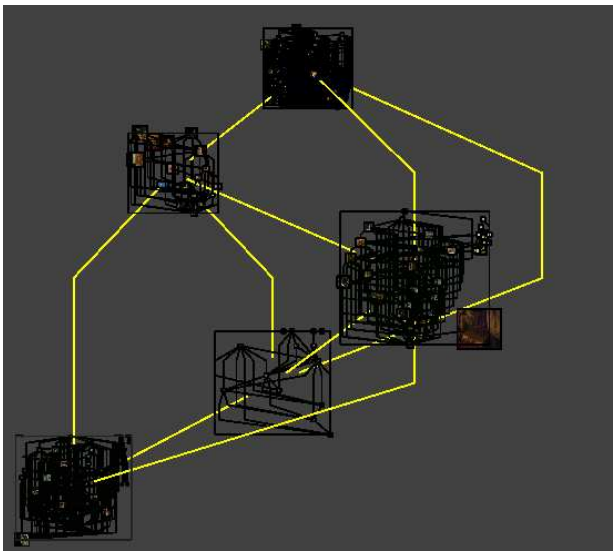

Figure 7. The five clusters.

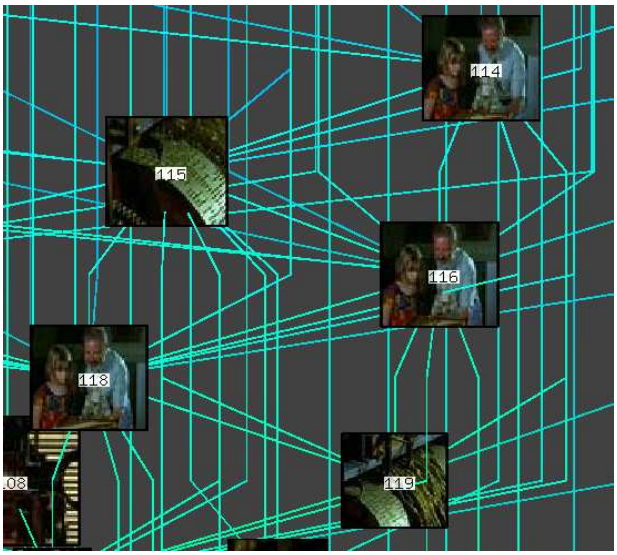

Figure 9. Navigating in a cluster.

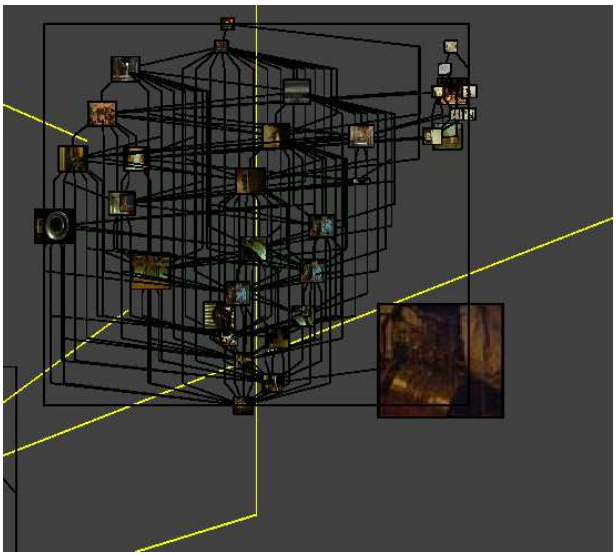

Figure 8. A cluster.

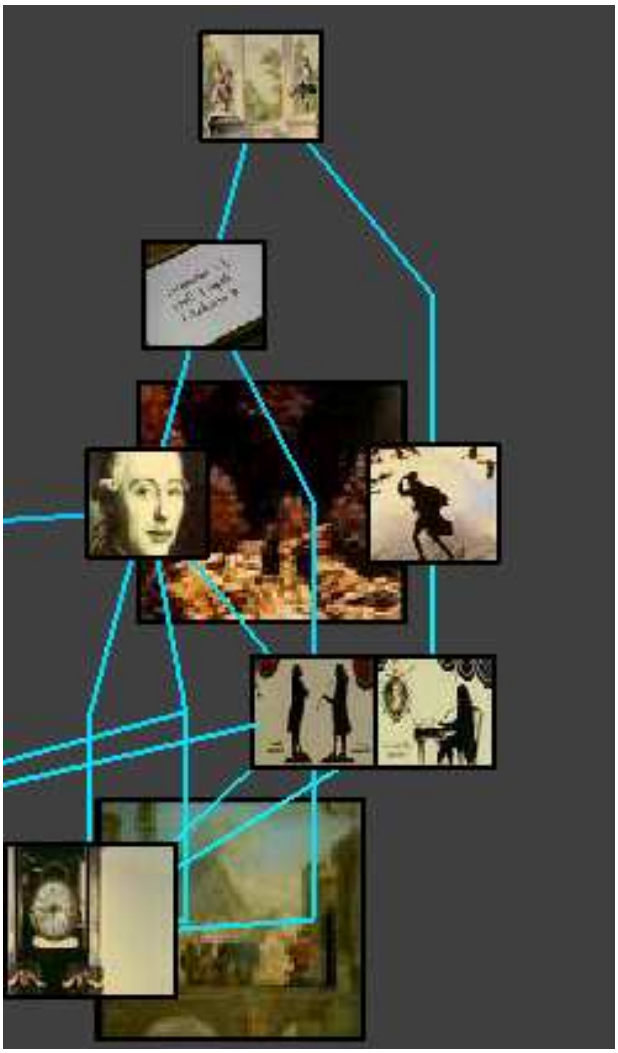

Figure 10. “Graphic shots” in Tympanon Movie