

Chapter 3

Multiple feature temporal modeling of visual contents

Most representations of contents used for video retrieval and high-level structuring are based on keyframes, and thus disregard the temporal component of video. However, image features can change their positions and values along a sequence of images. Sometimes, features can show temporal patterns, like in the case of temporal textures that were discussed in the previous chapter. These patterns can be found not only in motion features, but also in color and other visual cues. For instance, consider the images from fig. 3.1, which were taken from a sequence that shows a warning traffic light blinking. One out of every twenty images is shown. If we represented this sequence with one single keyframe, the blinking behavior would not be captured. Even two keyframes, for the “lit” and “unlit” states, would not represent the sequence correctly. In this case, the complete evolution of the color feature along the image sequence must be taken into account by the representation of contents, which leads us to modeling the behavior of features as temporal processes. In this chapter, a novel way of representing visual contents in video as temporal processes is presented. This way of modeling contents will allow us to combine multiple features in the same representation, as well as to evaluate their significance and the degree of dependency between them.

3.1 Markov chains

One of the simplest ways to describe a temporal process is a first order discrete Markov chain (MC). A discrete MC is a sequence X of ordered random variables X_t , $t \in [1, m]$, taking values in a state space $S = \{1, \dots, n\}$, which fulfills the Markov property:

$$P(X_t | X_{t-1}, \dots, X_1) = P(X_t | X_{t-1}) \quad (3.1)$$



Figure 3.1: One out of every twenty images of a warning traffic light blinking.

The PDF of a MC is given by:

$$P(X) = P(X_1) \prod_{t=2}^m P(X_t | X_{t-1}) \quad (3.2)$$

Figure 3.2(a) shows a graphical representation of this Markovian model. A MC is fully characterized by a n^2 -matrix of state transition probabilities T , where $T_{ij} = P(X_t = j | X_{t-1} = i)$. Given the definition of a MC, T has the following properties:

$$T_{ij} \geq 0, \quad \forall (i, j) \in S^2 \quad (3.3)$$

$$\sum_{j \in S} T_{ij} = 1, \quad \forall i \in S \quad (3.4)$$

The likelihood of a realization $x = \{x_1, \dots, x_m\}$, $x_i \in S$, of a MC with respect to a MC model Ψ is given by:

$$P(x | \Psi) = P(x_1 | \Psi) \prod_{t=2}^m P(x_t | x_{t-1}, \Psi) \quad (3.5)$$

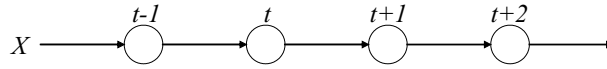
To simplify, we will omit the conditioning of the probabilities to the model Ψ , unless it may lead to confusion. The likelihood can also be expressed in terms of temporal cooccurrences:

$$P(x) = P(x_1) \prod_{(i,j) \in S^2} T_{ij}^{C_{ij|x}} \quad (3.6)$$

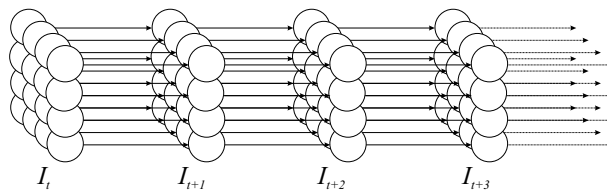
where $C_{ij|x}$ is the number of times that state j follows state i in x , i.e. the temporal cooccurrence of states i and j . We can take logarithms for simplicity and efficiency:

$$P(x) = P(x_1) \exp \left[\sum_{(i,j) \in S^2} C_{ij|x} \log T_{ij} \right] \quad (3.7)$$

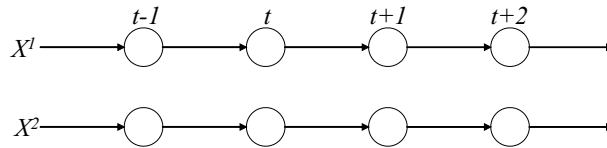
Equation (3.7) is the formulation of the MC likelihood as a Gibbs distribution with clique potentials $V_{ij} = \log T_{ij}$, as used by Fablet et al. in [18]. Note that



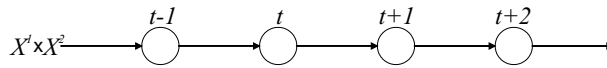
(a) Single Markov chain.



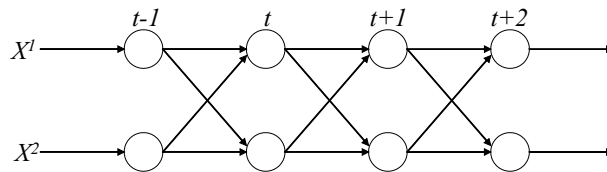
(b) 3-dimensional Markov random field for video contents representation based on Markov chains.



(c) Two independent Markov chains.



(d) Single Markov chain, where the random variables take values in the Cartesian product state space S^* .



(e) Two coupled Markov chains.

Figure 3.2: Graphical representations of the models discussed in the text, with 4 time steps unfolded.

the likelihood tends to 0 when the length of the chain grows. This fact causes an obvious computational problem for long chains, but also implies that the likelihoods of chains with different lengths cannot be directly compared. However, the temporal cooccurrences can be normalized, so that they would consider a chain of length 1, for instance. Given the cooccurrences $C_{ij|x}$ of states i and j in consecutive time steps in the chain x , we can express eq. (3.7) in terms of a normalized cooccurrence matrix \bar{C} , whose elements are:

$$\bar{C}_{ij|x} = \frac{C_{ij|x}}{\sum_{(i,j) \in S^2} C_{ij|x}}, \quad \forall (i,j) \in S^2 \quad (3.8)$$

These cooccurrences belong to a chain of length $\sum_{(i,j) \in S^2} \bar{C}_{ij|x} = 1$. Note that the normalized cooccurrence matrix is the joint distribution of X_t and X_{t-1} . Given a realization x :

$$P(X_t = j, X_{t-1} = i | X = x) = \bar{C}_{ij|x} \quad (3.9)$$

The likelihood of a length-normalized realization of a MC is then given by:

$$P(x) = P(x_1)^{\bar{C}_{i_1|x}} \exp \left[\sum_{(i,j) \in S^2} \bar{C}_{ij|x} \log T_{ij} \right] \quad (3.10)$$

where $\bar{C}_{i|x} = \sum_{j \in S} \bar{C}_{ij|x}$. That is, the frequency of state i in the observed chain x as in a histogram.

A second computational issue arises in the presence of observation noise. If there is a null transition probability in the model, a single noisy observation can make the whole likelihood of the realization drop to zero. A trivial example clarifies this statement. Let us consider a 2-state MC with transition probabilities $P(0|0) = 1$ and $P(1|0) = 0$, and a noiseless realization of this chain with $\bar{C}_{00} = 1$ and $\bar{C}_{01} = 0$. The likelihood in this case is 1. Consider now noisy observations with $\bar{C}_{00} = .99$ and $\bar{C}_{01} = .01$. The likelihood drops to 0. A solution to this problem is to introduce noise in the model itself, so that in our example we may have $P(0|0) = .99$ and $P(1|0) = .01$. The likelihood with noiseless observations in this case is .99, and with the noisy ones it only drops to .9455. Therefore, the model is much more robust to noisy observations.

For the estimation of parameters, given that the probabilistic model of a MC is defined as a fully-observed directed graphical model (i.e., there are no hidden variables), the maximum likelihood transition probabilities are directly obtained from an observation x as normalized frequencies (see appendix B for further details):

$$\hat{T}_{ij} = \frac{C_{ij|x}}{\sum_{j \in S} C_{ij|x}}, \quad \forall i \in S \quad (3.11)$$

X_t, X_{t-1}	$X_{t-1} = \text{ON}$	$X_{t-1} = \text{OFF}$
$X_t = \text{ON}$	49/75	1/75
$X_t = \text{OFF}$	1/75	24/75

Table 3.1: Joint probabilities for the blinking light example.

$X_t X_{t-1}$	$X_{t-1} = \text{ON}$	$X_{t-1} = \text{OFF}$
$X_t = \text{ON}$	0.98	0.02
$X_t = \text{OFF}$	0.04	0.96

Table 3.2: Transition probability matrix for the blinking light example.

A MC is the particular case of a 1-dimensional causal Markov random field (MRF). In order to apply this model to the representation of visual contents in a video sequence, we consider the video as a 3-dimensional MRF. Given a sequence of L images of size $M \times N$, the set of sites of the MRF is X_{uvt} , $(u, v, t) \in M \times N \times L$. The state of each site is given by the quantization of a scalar measure on a particular image feature. The set of cliques is formed by every pair of sites with the same spatial position and consecutive time instants:

$$\mathcal{C} = \{(I_t(x, y), I_{t-1}(x, y)), \forall (x, y, t)\} \quad (3.12)$$

where $I_t(x, y)$ is the pixel located at (x, y) at time t in image sequence I . Figure 3.2(b) shows the graphical model associated to this MRF. There is a random variable $X_t(x, y)$ attached to each pixel $I_t(x, y)$. Therefore, the PDF of the MRF X is:

$$P(X) = \prod_{x,y} P(X_1(x, y)) \prod_{t=2}^L P(X_t(x, y) | X_{t-1}(x, y)) \quad (3.13)$$

In this way, a global model of the temporal behavior of the feature in the sequence of images is obtained.

As an example of visual contents representation using this MC/MRF model, let us go back to the blinking light sequence from fig. 3.1. The sequence of images has the following characteristics:

- The light remains in the ON state for 2 seconds.
- The light is OFF during 1 second.
- The sequence was captured at 25 fps.

If only the image region occupied by the light is represented, the joint probabilities of X_t and X_{t-1} , and the corresponding transition probability matrix would be as shown in tables 3.1 and 3.2, respectively.

3.2 Measuring similarity of shots

The foundation of these Markovian models allows us to define measures of the similarity or the difference between two models Ψ_1 and Ψ_2 in terms of their probability distributions. Particularly, a dissimilarity measure can be defined as a likelihood ratio between the two models:

$$LR(\Psi_1||\Psi_2) = \frac{P(x^{\Psi_1}|\Psi_1)}{P(x^{\Psi_1}|\Psi_2)} \quad (3.14)$$

where x^{Ψ_1} is a sample from the distribution $P(X|\Psi_1)$. In practice, the same observations used for estimating the parameters of Ψ_1 are used to compute the likelihood ratio. Ψ_1 is formed by the maximum likelihood parameters for the observation x^{Ψ_1} . Therefore, $P(x^{\Psi_1}|\Psi_2)$ will always be smaller than $P(x^{\Psi_1}|\Psi_1)$, except for the case where Ψ_1 and Ψ_2 are exactly the same model. The likelihood ratio is in the range $[1, \infty]$. A similarity measure in the range $[0, 1]$ can then be obtained by inverting the likelihood ratio:

$$LR(\Psi_1||\Psi_2)^{-1} = \frac{P(x^{\Psi_1}|\Psi_2)}{P(x^{\Psi_1}|\Psi_1)} \quad (3.15)$$

Note that the likelihood ratio is not symmetric. For many applications, a symmetric similarity measure is needed. This can be obtained by combining the two opposed ratios:

$$S_{LR}(\Psi_1, \Psi_2) = \left[\frac{1}{2}(LR(\Psi_1||\Psi_2) + LR(\Psi_2||\Psi_1)) \right]^{-1} \quad (3.16)$$

Taking the logarithm of the likelihood ratio, we obtain the relative entropy between the two models, also known as cross entropy or Kullback-Leibler Divergence (KLD):

$$KLD(\Psi_1||\Psi_2) = \log \frac{P(x^{\Psi_1}|\Psi_1)}{P(x^{\Psi_1}|\Psi_2)} \quad (3.17)$$

KLD is a measure of the loss of accuracy to represent the observation x^{Ψ_1} using the distribution with parameters Ψ_2 instead of the real distribution given by the parameterization Ψ_1 . The KLD is greater than 0, and 0 when the two distributions are exactly the same. Again, it is not symmetric. A symmetric version can be obtained in a similar way as with the likelihood ratio:

$$S_{KLD}(\Psi_1, \Psi_2) = \frac{1}{2}(KLD(\Psi_1||\Psi_2) + KLD(\Psi_2||\Psi_1)) \quad (3.18)$$

Depending on the application, we can be interested on using likelihood ratios, KLD's or their symmetric versions to evaluate the similarity or dissimilarity of two shots represented as a Markovian process.

3.3 Coupling Markov chains to combine multiple features

Given a set F formed by f features, the goal is to combine their information into one unique temporal model. Having multiple features means that multiple observations (one per feature) are available at each site of the 3-dimensional MRF, i.e. at each pixel in the image sequence. Therefore, we can consider a set of MC's/MRF's $X = \{X^1, \dots, X^f\}$, one for each feature, with their own state spaces S^i , $\forall i \in [1, f]$, and couple them.

A first approach to couple multiple MC's is to consider that the chains attached to different features are independent. The graphical model is depicted in fig. 3.2(c). The PDF that represents this simple way of coupling is:

$$P(X) = P(X^1, \dots, X^f) = \prod_{i \in F} P(X^i) \quad (3.19)$$

where X^i is the sequence of random variables associated to feature i , and $P(X^i)$ is given by eq. (3.2). The main drawback of this approach is that the assumption of independence between features is not always true. A more realistic approach must consider possible dependencies and interactions that may exist between them.

In order to take into account these dependencies, the model shown in fig. 3.2(d) can be considered. This model is a single MC with a new state space S^* that is the Cartesian product of the state spaces of the features involved:

$$S^* = S^1 \times \dots \times S^f \quad (3.20)$$

In this case, all the features are tightly coupled. Therefore, this can be considered the optimal way of representing dependencies between them. However, the main disadvantage of this model is its computational cost. If the same number of quantization levels is assumed for all feature state spaces, the size of the transition matrix will be n^{2f} . For $n = 16$ states, $f = 2$ features and double precision floating point elements, the transition probability matrix has size 512 KB. For $f = 3$, it grows up to 128 MB, which is completely unaffordable in terms of storage size and computation time.

Given that the Cartesian-product model can be considered the optimal way of representing features between dependencies, it can be used as a starting point to obtain other models with less cost by assuming independencies between the random variables involved. First, we can assume independence between the features at the same time instant, while they still depend on all the features at the previous time. This assumption is expressed by the following factorization of the transition probabilities:

$$P(X_t^1, \dots, X_t^f | X_{t-1}^1, \dots, X_{t-1}^f) = \prod_{i \in F} P(X_t^i | X_{t-1}^1, \dots, X_{t-1}^f) \quad (3.21)$$

The joint PDF for chains of length m is then given by:

$$P(X) = P(X_1^1, \dots, X_1^f) \prod_{t=2}^m P(X_t^1, \dots, X_t^f | X_{t-1}^1, \dots, X_{t-1}^f) \quad (3.22)$$

$$= \prod_{i \in F} P(X_1^i) \prod_{t=2}^m P(X_t^i | X_{t-1}^1, \dots, X_{t-1}^f) \quad (3.23)$$

This new model, depicted in fig. 3.2(e), is called Coupled Markov Chains (CMC). In this case, the size of the transition probability matrix is $f n^{f+1}$. For $n = 16$, $f = 2$ and double precision, the size is 64 KB. For $f = 3$, it is 192 KB, which is 4 times smaller than the Cartesian product model with $f = 2$. The problem is to know whether these reductions of the cost in time and space turn into a loss of representation accuracy.

3.4 Structure learning

The problem of structure learning consists of finding the optimal configuration of a model with respect to some criteria. In our case, the structure of the model is given by the dependencies between the random variables involved. These dependencies are represented by directed links in a graph where the nodes are the variables. Our problem is how to decide what links can be removed, that is, what variables are independent from each other.

The single MC is the simplest model with dependencies between variables. Given this model structure and its likelihood from eq. (3.10), the log-likelihood of a length-normalized observation is given by:

$$\mathcal{L}\{P(x)\} = \sum_{(i,j) \in S^2} \bar{C}_{j,i|x} \log T_{ij} \quad (3.24)$$

For simplicity, the following notation will be used in this section:

$$P(j, i) = P(X_t = j, X_{t-1} = i) \quad (3.25)$$

$$P(j|i) = P(X_t = j | X_{t-1} = i) \quad (3.26)$$

$$P(i) = P(X_t = i) \quad (3.27)$$

Remembering that the normalized cooccurrence matrix of a realization x is actually a joint distribution $\bar{C}_{ij|x} = P(X_t = j, X_{t-1} = i | X = x)$, the log-likelihood can be rewritten as follows:

$$\mathcal{L}\{P(x)\} = \sum_{(i,j) \in S^2} P(j, i|x) \log P(j|i) \quad (3.28)$$

When the observation x is the same that we used to estimate the parameters of the model, eq. (3.28) is minus the conditional entropy of X_t given X_{t-1} , $H[P(X_t|X_{t-1})]$. We will omit the conditioning on x from $P(j, i|x)$, as this fact will be assumed from now on unless it may lead to confusion. The conditional entropy is a measure of the amount of information that knowledge about the conditioning variable provides about a certain random variable. Let us consider the case where this knowledge about the conditioning variable is not available. We have a new set of transition probabilities $P'(X_t|X_{t-1})$, where X_t does not depend on the value of X_{t-1} , so that $P'(X_t|X_{t-1}) = P(X_t)$. The log-likelihood for this new distribution is measured as:

$$\mathcal{L}\{P'(x)\} = \sum_{(i,j) \in S^2} P(j, i) \log P'(j|i) \quad (3.29)$$

$$= \sum_{i \in S} P(i) \log P(i) \quad (3.30)$$

We know that $P(X_t)$ may contain less information about the random variable X_t because we do not have the knowledge about a second random variable (X_{t-1}) that may be important. We can compute this loss of information of the simplified distribution $P'(X)$ with respect to the true distribution $P(X)$ as a difference of their log-likelihoods, or entropies:

$$D(P||P') = \mathcal{L}\{P(X)\} - \mathcal{L}\{P'(X)\} \quad (3.31)$$

$$= -H[P(X_t|X_{t-1})] + H[P(X_t)] \quad (3.32)$$

If the loss of information of $P'(X)$ with respect to $P(X)$ is close to zero, then the two distributions are very similar and there was no real dependency of X_t on X_{t-1} . In that case, considering the temporal dependencies is not necessary, which can be graphically expressed as the removal of the link from X_{t-1} to X_t . A simple manipulation of this expression leads us to the definition of relative entropy (also known as cross entropy or Kullback Leibler distance). Relative entropy is a measure of the distance between distributions that has been used for structure learning in Bayesian networks [40]. It measures the loss of information of a probability distribution with respect to the true distribution, and is defined in our case as:

$$D(P||P') = \sum_{(i,j) \in S^2} P(j, i) \log \frac{P(j|i)}{P(j)} \quad (3.33)$$

Equation (3.33) is the loss of information when we represent a sequence using a simple histogram, instead of considering a first-order temporal dependency.

Going back to the Cartesian product and the CMC model structures, let us consider their respective conditional entropies. The case of $f = 2$ will be considered for simplicity:

$$H[P(X_t^1, X_t^2 | X_{t-1}^1, X_{t-1}^2)] = - \sum_{i,j,k,l} P(i, j, k, l) \log P(i, j | k, l) \quad (3.34)$$

$$H[P'(X_t^1, X_t^2 | X_{t-1}^1, X_{t-1}^2)] = - \sum_{i,j,k,l} P(i, j, k, l) \log [P(i|k, l)P(j|k, l)] \quad (3.35)$$

where the indexes i, j, k, l are attached to the random variables $X_t^1, X_t^2, X_{t-1}^1, X_{t-1}^2$, respectively. The relative entropy of the CMC structure with respect to the Cartesian product structure is:

$$D(P||P') = \sum_{i,j,k,l} P(i, j, k, l) \log \frac{P(i, j | k, l)}{P(i|k, l)P(j|k, l)} \quad (3.36)$$

which is the definition of conditional mutual information of X_t^1 and X_t^2 given X_{t-1}^1 and X_{t-1}^2 . Equation (3.36) provides a measure of the amount of information lost by the assumption of independence that turns the Cartesian product model structure into the CMC structure, and thus the loss of accuracy in representing the relationships between image features. Equation (3.36) also lets us obtain the optimal transition probabilities for the CMC structure in terms of minimal mutual information with respect to the Cartesian product structure:

$$\hat{P}(i|k, l) = \frac{\sum_j P(i, j, k, l)}{\sum_i \sum_j P(i, j, k, l)} \quad (3.37)$$

$$\hat{P}(j|k, l) = \frac{\sum_i P(i, j, k, l)}{\sum_j \sum_i P(i, j, k, l)} \quad (3.38)$$

Details are given in appendix C.

Further independencies can be assumed on the CMC model, which are graphically represented by removing links from the structure in fig. 3.2(e). Every independency assumption turns into a reduction of the cost in time and space of the model. Table 3.3 summarizes the model structures that are considered in this work, and shows the size of the transition probability matrix for 2 features and different number of quantization levels (states).

A space of feasible model structures can be built as a directed graph, where the nodes are model structures, and the links represent independency assumptions. Relative entropy as a measure of information loss has some properties that will allow us to search this space of feasible model structures more efficiently:

Property 1 Chain rule: *The length of the path from P to P''' is the sum of partial paths, and does not depend on what link is removed first.*

Let us consider the following distributions from fig. 3.3 (top and middle-left):

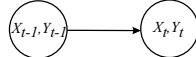
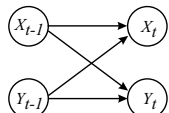
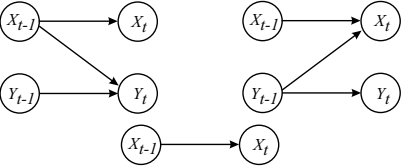
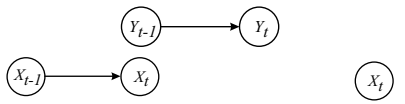
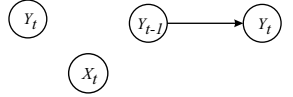

Structure	Graphical model	Size	$n = 16$	$n = 8$	$n = 4$
Cartesian product (CP)		n^4	65536	4096	256
Full CMC (FC)		$2n^3$	8192	1024	128
Partial CMC (PC)		$n^3 + n^2$	4352	576	80
Independent chains (I)		$2n^2$	512	128	32
Partial temporal (PT)		$n^2 + n$	272	72	20
Histograms (H)		$2n$	32	16	8

Table 3.3: Cost of different model structures with 2 features.

$$P(X, Y, Z) = P(X|Y, Z)P(Y)P(Z) \quad (3.39)$$

$$P'(X, Y, Z) = P(X|Y)P(Y)P(Z) \quad (3.40)$$

$$P''(X, Y, Z) = P(X|Z)P(Y)P(Z) \quad (3.41)$$

$$P'''(X, Y, Z) = P(X)P(Y)P(Z) \quad (3.42)$$

In this case, the loss of information of $P'''(X, Y, Z)$ w.r.t. $P(X, Y, Z)$ is:

$$\begin{aligned} D(P||P') &= \sum_{x,y,z} P(x, y, z) \log \frac{P(x|y, z)}{P(x|y)} \\ &= \sum_{x,y,z} P(x, y, z) \log \frac{P(x|y, z)P(x|y)}{P(x|y)P(x)} \\ &= \sum_{x,y,z} P(x, y, z) \left[\log \frac{P(x|y, z)}{P(x|y)} + \log \frac{P(x|y)}{P(x)} \right] \\ &= \sum_{x,y,z} P(x, y, z) \log \frac{P(x|y, z)}{P(x|y)} + \sum_{x,y,z} P(x, y, z) \log \frac{P(x|y)}{P(x)} \\ &= D(P||P') + D(P'||P''') \end{aligned} \quad (3.43)$$

In the same way, using $P(x|z)$ instead of $P(x|y)$, we reach:

$$D(P||P''') = D(P||P'') + D(P''||P''') \quad (3.44)$$

Given that relative entropies are always positive or zero, this also means that the loss of information due to successive link removal increases monotonically. \square

Property 2 Locality: *For multiple chains, the total relative entropy is the sum of local relative entropies.*

Given that our structures are always defined by directed acyclic graphs (DAG's), the relative entropies between them can be computed locally. Consider the CMC and the independent MC's models from fig. 3.4, with their respective probability distributions P_C and P_I . The relative entropy between them is:

$$\begin{aligned} D(P_C||P_I) &= \sum_{i,j,k,l} P(i, j, k, l) \log \frac{P(i|k, l)P(j|k, l)}{P(i|k)P(j|l)} \\ &= \sum_{i,k,l} P(i, k, l) \log \frac{P(i|k, l)}{P(i|k)} + \\ &\quad + \sum_{j,k,l} P(j, k, l) \log \frac{P(j|k, l)}{P(j|l)} \end{aligned} \quad (3.45)$$

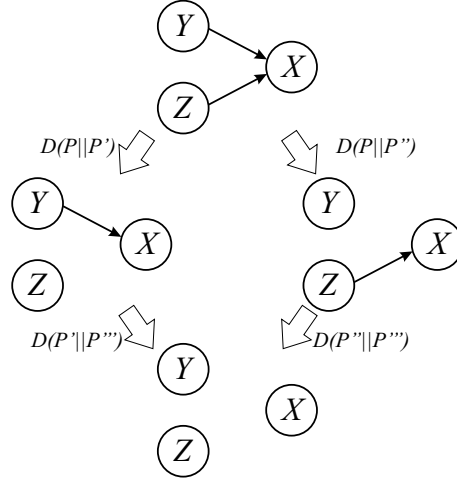


Figure 3.3: Example of the process of removing dependencies between variables (links) from the structure. First, X depends on Y and Z (top). When one link is removed, X only depends on one variable, either Y (left) or Z (right). Finally, X is completely independent (bottom).

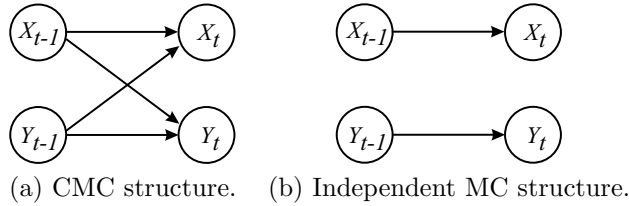


Figure 3.4: Model structures used to illustrate the locality property.

where the indexes i, j, k, l are attached to the random variables $X_t, Y_t, X_{t-1}, Y_{t-1}$, respectively. \square

Costs can be attached to the links of the graph of feasible structures in terms of the loss of information given by the relative entropy between model structures. Given that relative entropy is a subtraction of two entropies, costs can also be attached to the nodes in terms of the amount of information that the models contain, as given by their entropies, and then compute relative entropies from them. Depending on the number of feasible model structures considered, this can turn into a reduction of the computation needed to obtain the costs of the paths in the space of structures. Let us suppose that all the links in the model are allowed to be removed, and the original structure has L links. In that case, the number of feasible structures (nodes in the space of structures) would be:

$$N_{nodes} = \sum_{i=0}^L \binom{L}{i} = 2^L \tag{3.46}$$

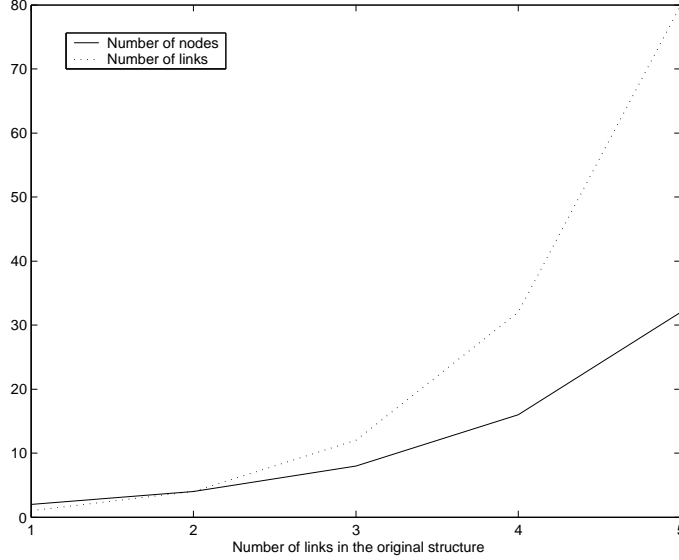


Figure 3.5: Number of nodes and links of the space of feasible model structures, as a function of the number of links in the original model structure.

However, the number of links in the graph of feasible structures would be:

$$N_{links} = \sum_{i=0}^{L-1} (L-i) \binom{L}{i} = L2^{L-1} \quad (3.47)$$

Figure 3.5 shows the number of nodes and links of the space of feasible model structures as a function of the number of links in the original model structure. For 2 or more links in the original structure, it is more convenient to compute conditional entropies as costs for the nodes instead of relative entropies for the links. However, we must note that the properties of relative entropy can allow us to save computations as well.

Different criteria can be used to decide when a reduced structure is the optimal with respect to the true one. We can specify an acceptance threshold on the relative entropy that will determine the loss of information that we want to afford. Given that, after the chain rule property, the loss of information increases monotonically with the length of the path in the space of structures, once the threshold is exceeded the rest of the path can be automatically discarded.

Sequence name	Domain	Number of shots
News	Newscast	121
Soccer	Sports	232
Friends	Sit-com	205

Table 3.4: Details of the video sequences used in the content-based video retrieval experiments.

3.5 Information loss in practice

This section shows the practical side of the theory developed so far in this chapter. For this purpose, the loss of information of the model structures discussed in previous sections has been computed using test video sequences from different domains. Table 3.4 shows details of these sequences.

Video shots were described using one color and one motion feature. The color feature was the Hue component of the HSV color space. This measure represents the identity of the color. The motion feature was the normal flow. The normal flow is the projection of the optical flow in the direction of the gradient, and thus provides a measure of the amount of motion. Given the image intensity function I , the normal velocity at pixel p is given by:

$$v_n(p) = \frac{-I_t}{\|\nabla I(p)\|} \quad (3.48)$$

where I_t is the temporal derivative of the intensity function I , which can be approximated by a simple finite difference. Fablet et al. [18] defined a more reliable measure, by averaging the normal velocity over a small window with weights in terms of the gradient:

$$v_{obs}(p) = \frac{\sum_{q \in \mathcal{W}(p)} \|\nabla I(q)\|^2 |v_n(q)|}{\max(G^2, \sum_{q \in \mathcal{W}(p)} \|\nabla I(q)\|^2)} \quad (3.49)$$

where $\mathcal{W}(p)$ is a small window centered on p and G^2 is a predetermined constant related to the noise level in uniform areas. Both color and motion features were quantized into 16 possible states.

Previously in this chapter, we have stated that the Cartesian product structure can be considered as the optimal way of representing the possible causal dependencies between features (color and motion in this case) that may exist. From the Cartesian product structure, successive independency assumptions were considered in order to generate the space of feasible structures shown in fig. 3.6. The costs shown in this figure correspond to averaging over the shots of the Friends test sequence. In the case of partial models, either in the coupled or in the temporal sense, we differentiate between the best and the worst partial structure. The best structure is the one with the lowest loss of information, and vice versa. Note that being the best or the worst

partial structure is decided for each different shot.

Figure 3.7 shows the average loss of information of the different model structures over all the shots of the different domains from the test sequences. The three domains of video contents that are considered are shown in this plot, in order to perceive possible differences in the behavior of the structures. In general, we observe that the most important loss of information appears when temporal links within the same feature are removed. Particularly, in the Friends and the Soccer sequences the temporal information of one of the features (color in some shots, motion in some others) is even more relevant than the temporal information of the other. In the News sequence, the temporal information of both features appears to be equally relevant. In any case, the information provided by the crossed links is less relevant than the temporal information within the same features. However, a significant loss of information can be noticed, especially in the News sequence. Particularly, this happens when the shot contains objects with different color and motion features than the rest of the scene, like the wavy flag shown in fig. 3.8(a). In this case, information about the relationship between the white and blue colors in the flag and its wavy motion is contained in the crossed links. Another example is shown in fig. 3.8(b), where the motion of Phoebe laughing is related to her skin and hair colors, which are very different to the rest of the scene. It is also important to note that the loss of information of the fully coupled structure with respect to the Cartesian is very small despite the large reduction in storage size.

A significant conclusion reached after the previous observations is that the accumulation of static image descriptors along the sequence is not a good approach to representing dynamic contents. For instance, an accumulated color histogram will clearly not capture the behavior of the blinking light from fig. 3.1. A simple MC using color observations will provide much more information about the blinking behavior.

Figure 3.9 allows us to better observe this behavior and the differences between the domains. In this case, the average amount of information that the different model structures contain about our test video data in terms of average likelihoods is shown. We can see that the amount of information drops from the fully coupled to the independent chains structures faster in the Friends and the News sequences than in the Soccer sequence.

3.6 Information loss vs. Contents representation accuracy

The goal of the experiments shown in this section is to analyze the effect of the loss of information caused by the use of a simplified model structure on its accuracy to represent visual contents from videos. For this purpose, content-based video shot retrieval experiments have been conducted on the same test sequences and model configurations used above. The likelihood ratio from eq. (3.14) was used as dissimilarity measure. The results have been compared to the measures of information loss computed from the theory in the previous section.

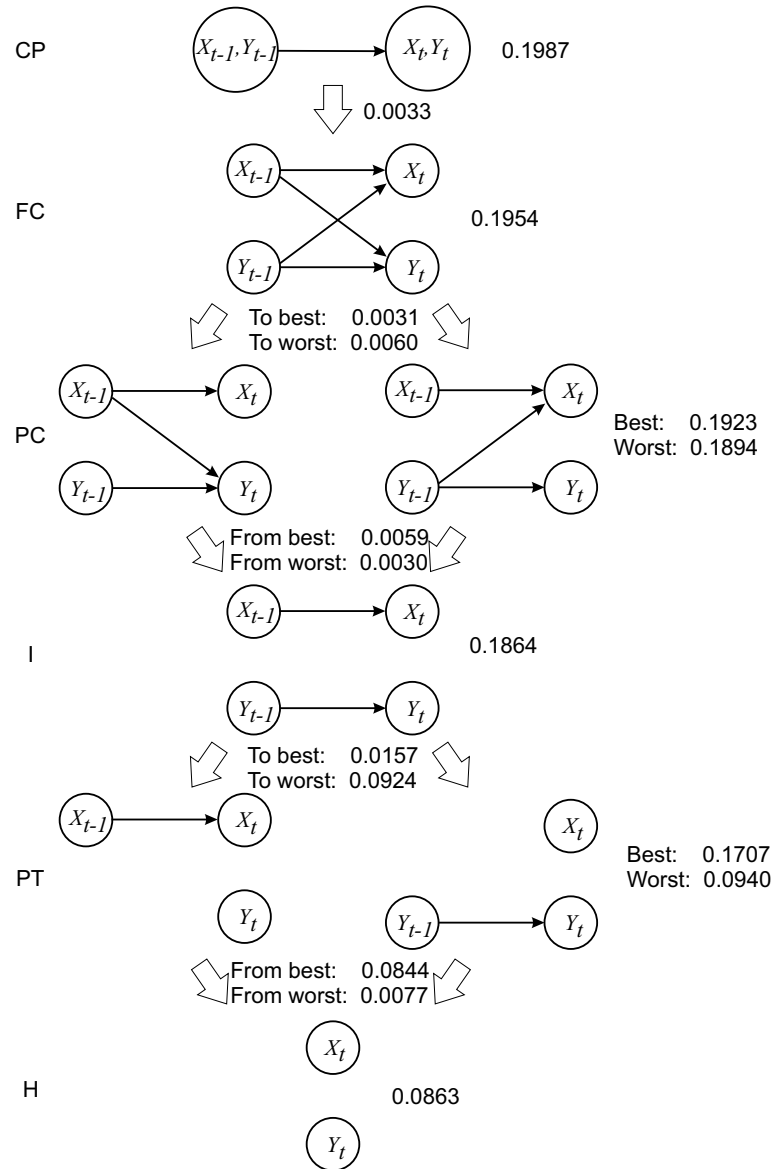


Figure 3.6: Space of feasible model structures considering two features, and the paths between them defined by successive independency assumptions. More configurations exist, but are not considered here. The costs correspond to the Friends sequence. Model structure identifiers are taken from table 3.3.

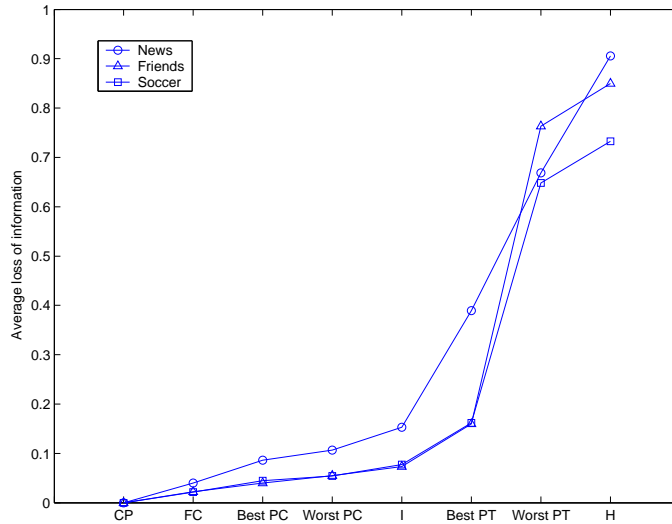
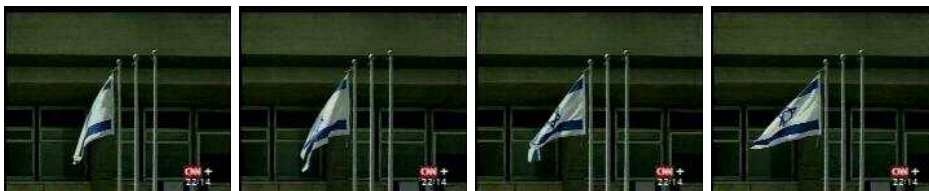


Figure 3.7: Loss of information of different model structures with respect to the Cartesian product structure given by relative entropies, and averaged over all the shots of the test sequences. Model structure identifiers are taken from table 3.3.



(a) Wavy flag.



(b) Laughing Phoebe.

Figure 3.8: Images from shots where the crossed links between color and motion features are very significant for representing their relationship in the scene.

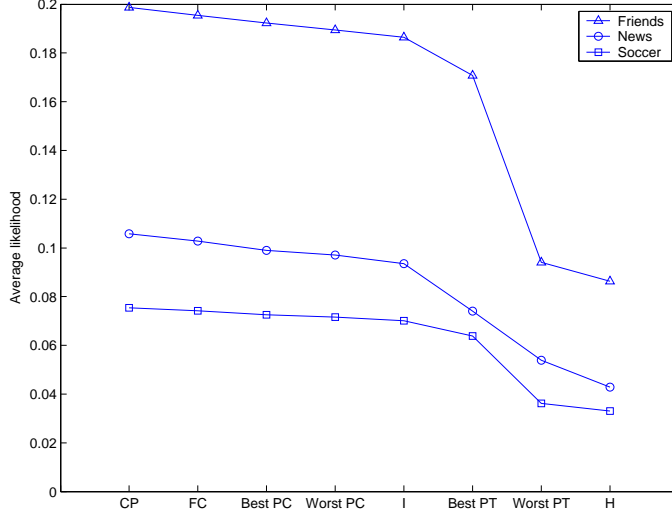


Figure 3.9: Amount of information of different model structures given by their likelihoods, and averaged over all the shots of the test sequences. Model structure identifiers are taken from table 3.3.

In order to evaluate the quality of the retrieval, we use a measure based on the classical precision:

$$Precision = \frac{N_{correct}}{N_{retrieved}} \quad (3.50)$$

where $N_{correct}$ is the number of correct results within the $N_{retrieved}$ shots retrieved. Correct shots are determined by the retrieval results obtained using the Cartesian product structure, given that it has been considered as the optimal way of representing contents. When $N_{retrieved}$ is relatively large with respect to the total number of shots in the database (N_{total}), the precision is not meaningful by itself. Even a random selection of shots in the database would have a high precision. It is more meaningful to measure the improvement of the precision over the precision of a random selection. This measure is given by Cohen's κ statistic:

$$\kappa = \frac{Precision - Precision_{random}}{1 - Precision_{random}} \quad (3.51)$$

where $Precision_{random}$ is the expected value of precision when the results are selected randomly, and is given by (see appendix D):

$$Precision_{random} = \sum_{K=\max(0,2M-N)}^M \frac{K}{M} P(K) \quad (3.52)$$

where $M = N_{retrieved}$ and $N = N_{total}$.

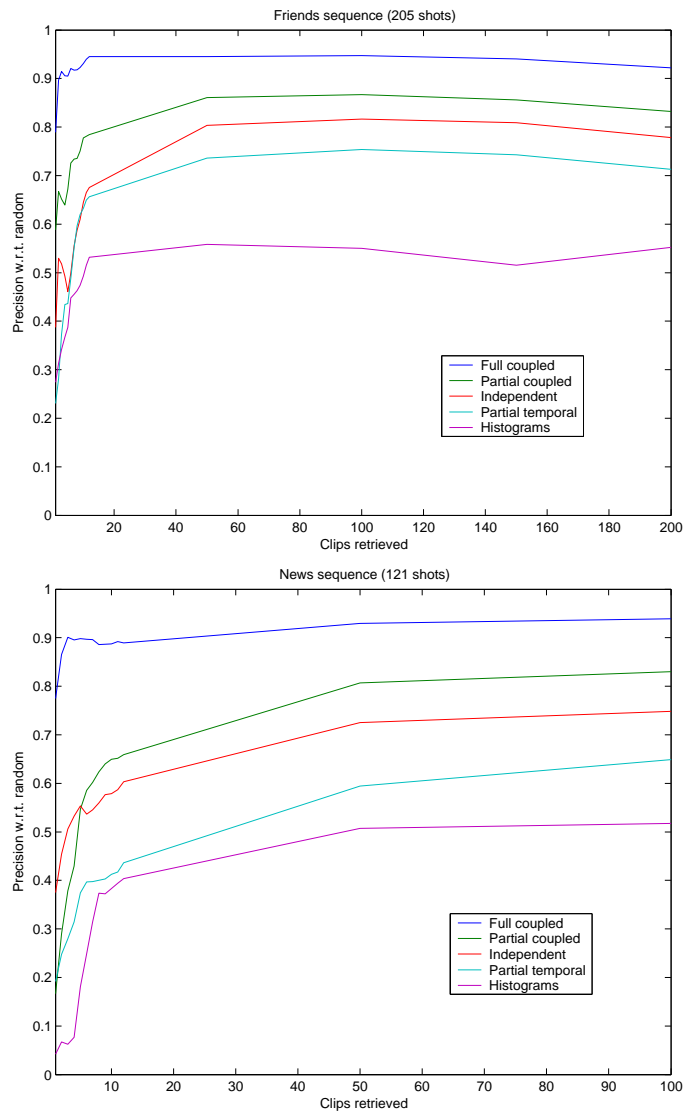


Figure 3.10: Evaluation of retrieval using different model structures with our three test videos. The ground-truth is the retrieval results using the Cartesian product model structure. The measure is given as the improvement of precision over a random selection of shots. Measures are taken for $N_{retrieved} = \{1..12, 50, 100, 150, 200\}$.

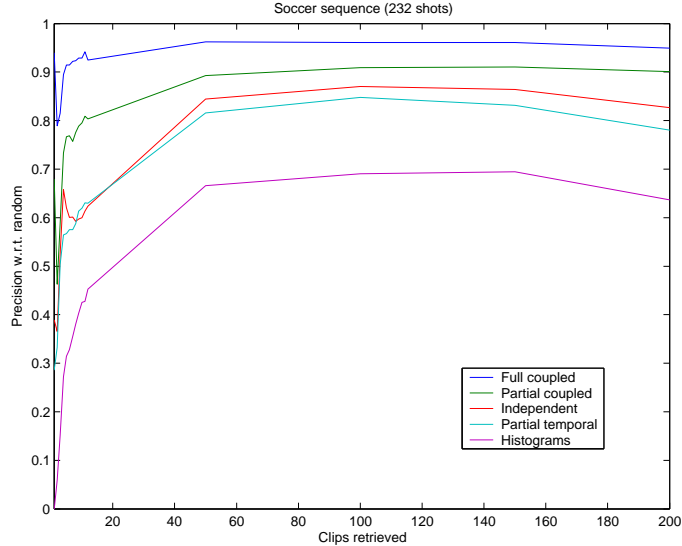


Figure 3.10: (Continued).

Figure 3.10 shows the κ statistic of the precision for different values of $N_{retrieved}$ in the three videos used in our experiments. The results are averages using the leave-one-out method for all the shots in the database. The results obtained are consistent with the previously computed loss of information. That is, the higher loss of information of the structure, the less accurate retrieval results are obtained. Particularly, results show that the fully coupled structure is a very good approximation of the ideal Cartesian product structure (over 90% of correct clips retrieved in all cases). Also, the performance of the simple histograms is very low, which means that the temporal evolution of features is relevant for describing visual contents that may have dynamic behaviors. Therefore, accumulating static image descriptors through several images in a sequence is not an appropriate approach to represent temporal sequences.

It is important to observe that the loss of performance of simplified model structures is directly related to their loss of information about the original data. For instance, note the large difference of performance in the Friends sequence between the partial temporal structure and the histograms in fig. 3.10(top), which is related to the large loss of information between those structures shown in figs. 3.7 and 3.9. This similarity between the relative differences in performance and the loss of information can also be noticed in the News and the Soccer sequence. We cannot establish a numerical or analytical relationship, but there is an obvious qualitative dependency between them.

3.7 Characterization of “activity”

Figure 3.9 lets us observe an important fact. The average likelihoods, i.e. amount of information, of different domains of video contents have different energy levels. Initially, this energy level seems to be related to the amount of activity in the videos. The Friends sequence is mainly shot using close-ups and medium shots for the dialogues, which are the main resource in sit-coms. Close-ups and camera operation are very common resources used to communicate high activity, particularly in Latin-American soap-operas. On the other hand, the Soccer sequence is mainly composed by wide-angle shots that follow the progress of the game. Activity is really significant only when a goal is scored. Besides, sit-coms offer a larger variety of colors, due to the use of different settings and clothing. In the case of soccer, the color is mainly green from the playing field, and the colors of the team jerseys do not change during the game. A higher variety of colors combined with closer camera shots convey a higher sense of activity to the viewer (this is probably the reason why watching soccer can be boring if the viewer is not emotionally involved with one of the teams). The domain of News is a mixture of both. There are close-ups for interviews, panoramic views to illustrate events, and a relatively assorted set of colors. A measure of activity can provide one more intermediate-level descriptor, which is at the same time implicit in the CMC representation.

In order to assess the relationship between the average amount of information contained in the CMC models and the average activity of the videos, we have conducted experiments in a different domain of video contents: movie trailers. An instance of this domain is specifically produced to convey a feeling about the movie, so that the target audience will be teased to watch it. Therefore, the trailer of an action movie will have lots of action, and thus activity, while the activity of a trailer from a romantic movie will be much lower. Moreover, it is easy to obtain ground-truth about the semantic categorization of movies from different Internet sites like the Internet Movie Database (IMDb) [38] and Movies.com [50].

The shots of 32 movie trailers were modeled using CMC’s as described in previous sections. The same features and quantization levels that generated the plot from fig. 3.9 were used in this case. The measure of activity was computed as the average likelihood for all the shots of the trailer. Results are shown in table 3.5. Activity goes from lower (top) to higher (bottom). The semantic categories assigned by IMDb and Movies.com are also specified in columns 2 and 3, respectively. We can almost draw a clear line between the movies in the categories Drama/Comedy/Romance, whose activity is low, and the movies in Action/Adventure/Thriller, with higher activity. It is also interesting to observe how different trailers of the same movie are intended to tease different target audiences. This is the case of Star Wars Episode 2 and its trailers subtitled Breathing, Forbidden Love, Clone War and Mystery. The trailers with higher activity mainly show action scenes from the movie, while the trailers with lower activity show romantic and dramatic situations.

Vasconcelos and Lippman also addressed in [76] the genre classification problem using measures of activity and shot length. Semiotics also relates shot length to a

Movie title	IMDb categories	Movies.com categories	Activity
1. The lady and the duke	Drama	Drama, historical	1,3941
2. Never again	Comedy, romance	Comedy, romance	1,8863
3. Signs	Sci-Fi, drama, thriller, fantasy	Thriller, supernatural	1,9559
4. The importance of being earnest	Drama, comedy, romance	Comedy, historical	1,9619
5. Chelsea walls	Drama	Drama	2,0144
6. The Bourne identity	Action, thriller	Spy, thriller	2,0381
7. About a boy	Drama, comedy	Comedy, romance	2,0661
8. The son of the bride	Drama, comedy	Drama, comedy	2,0663
9. Star Wars Ep. 2: Attack of the clones (Breathing)	Sci-Fi, adventure, action	Sci-Fi, action	2,0900
10. Happy times	Drama, comedy	Comedy, romance	2,0931
11. Cherish	Drama, comedy, thriller	Crime, romance	2,1234
12. One hour photo	Thriller	Thriller	2,1445
13. Austin Powers in Goldmember	Comedy	Comedy	2,1675
14. CQ	Drama	Comedy, drama	2,2005
15. Cinema paradiso	Drama, romance	Foreign, drama	2,3654
16. Spirit: stallion of the cimarron	Animation, western, family	Animated, western	2,4039
17. Halloween resurrection	Horror, thriller	Horror, thriller	2,4569
18. Star Wars Ep. 2: Attack of the clones (Forbidden love)	Sci-Fi, adventure, action	Sci-Fi, action	2,4605
19. Star Wars Ep. 2: Attack of the clones (Clone war)	Sci-Fi, adventure, action	Sci-Fi, action	2,4611
20. Spy kids 2	Adventure, family	Children’s, fantasy	2,6405
21. Undercover brother	Action, comedy	Comedy, martial arts	2,6546
22. Reign of fire	Sci-Fi, action	Action, fantasy	2,6901
23. Men in black II	Sci-Fi, fantasy, action, comedy	Sci-Fi, action	2,6985
24. The believer	Drama	Drama, biography	2,7547
25. Unfaithful	Drama, thriller	Thriller, erotic	2,8097
26. Spider-man	Action, fantasy, sci-fi, adventure, romance, thriller	Action, fantasy	2,8907
27. Star Wars Ep. 2: Attack of the clones (Mystery)	Sci-Fi, adventure, action	Sci-Fi, action	2,9824
28. The sum of all fears	Drama, action, adventure, thriller	Action, thriller	3,0496
29. Die another day	Action	Action, spy	3,0889
30. Enough	Drama, thriller, action	Thriller, crime	3,1002
31. Insomnia	Crime, thriller	Crime, thriller	3,1228
32. Minority report	Sci-Fi, thriller, action	Sci-Fi, action	3,1520

Table 3.5: Activity of trailers from different movies. High activity is associated to action, adventure and thriller, while low activity is mainly associated to drama, comedy and romance. The classes Drama/Comedy/Romance and Action/Adventure/Thriller can be practically separated.

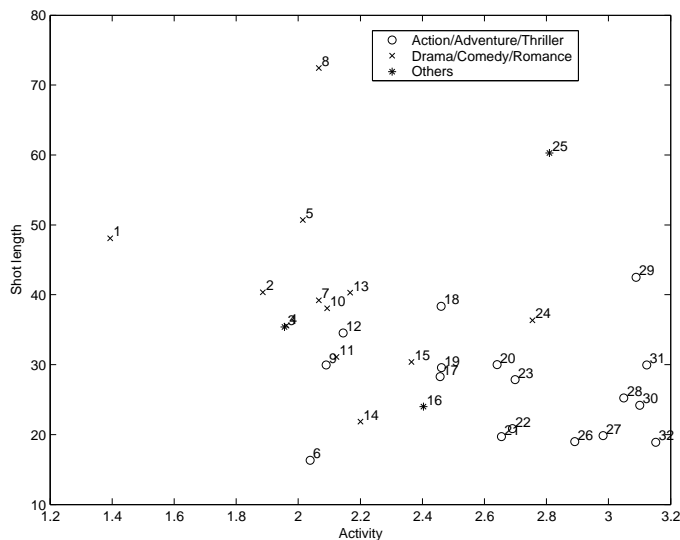


Figure 3.11: Plot of shot length vs. activity of the movie trailers from table 3.5.

sense of dynamism and action. They classified movie trailers into three categories: Romance/Comedy, Action and Other. Considering also shot length in our set of movie trailers from table 3.5, their distribution in this new 2-feature space is shown in fig. 3.11. Movies in other IMDb/Movies.com categories (animation, western) or containing a mixture of our classes (drama AND thriller) have been attached to the class Others. We observe that shot length is not contributing positively to the separability of the classes Action/Adventure/Thriller and Drama/Comedy/Romance. This was also noticeable in the work by Vasconcelos and Lippman, when they used tangent distance to compute their activity measure. In that case, the lines they draw to separate their classes are almost vertical, which means that the shot length axis contains little additional information. Shot length is significant in the case of trailer number 6 “The Bourne Identity”, an Action/Thriller movie whose trailer has low activity measure, but also the lowest average shot length. In this particular case, shot length provides the clue for a correct classification.

3.8 Summary

This chapter has introduced a novel way of representing visual contents in video. Image sequences are modeled as temporal processes using Markov chains/Markov random fields. Sites are associated to pixels, and low-level image features like color or motion are observed at each site. Multiple features are combined into the same representation by coupling their associated Markov chains. In this work, the Cartesian product state space model has been considered as the optimal way of coupling multiple features. However, when the number of features grows, the storage size and

processing time of these models becomes prohibitive. Independencies between the random variables involved in the model can be assumed in order to reduce its cost. Relative entropy has been used as a measure that allows us to evaluate the amount of information lost in the representation due to these independency assumptions, which has led us to develop a method to automatically determine the best model structure in terms of minimum information loss. Experiments have been conducted to establish a relationship between the loss of information of simplified model structures and their accuracy to represent visual contents in terms of content-based video retrieval. After the experiments, we reach the following conclusions:

- The CMC model structure is an accurate approximation to the Cartesian product structure, and its cost is enormously reduced.
- The largest loss of information occurs when the temporal dependency within the same feature is removed. Therefore, it is not a good approach to represent visual contents with dynamic behaviors by accumulating static image descriptions of the images in the sequence.
- The crossed links between features are important when there are elements in the scene with a clear dependency between them, like in the examples of the wavy flag and laughing Phoebe.
- For content-based video retrieval, the higher loss of information of the model structure we are using, the less accurate retrieval results are obtained.
- The loss of performance of simplified model structures is directly related to their loss of information about the data that they are representing.
- The amount of information contained in the CMC model structure gives a measure of the activity of the shot. Important semantic information like genre can be inferred from the activity of a video.

Chapter 4

Semantic analysis of video contents using coupled Markov chains

This chapter is focused on the application of the multiple feature temporal models developed in the previous chapter. In the same way a color histogram summarizes certain information about the appearance of an object, and in this way captures some of its semantics, the CMC-based representation is a summary of multiple low-level image features, their temporal behaviors, and the possible relationships of dependency that may exist between them. Thus, the intermediate-level semantics captured by this modeling must be even more relevant for higher level video structure analysis than simply using color summaries. Particularly, the CMC representation can be used to obtain an intermediate-level semantically meaningful clustering of video shots that will help us to extract higher level structures by defining simple rules in the particular case of news videos. This chapter also deals with object localization in image sequences and shot boundary detection.

4.1 Object detection in video

Object detection in video sequences has a fundamental importance for indexing and annotation. Unconstrained video is a particularly challenging domain where very few assumptions about the objects and the scene can be done. Clutter, occlusions and all kind of variations of the objects can be found.

Current approaches deal with video frames as static images and apply static object detection techniques on them. Appearance-based approaches (like Murase and Nayar in [51]) have been proposed instead of model-based in order to obtain more general and easily trainable systems. Many different features have been used to represent the appearance of objects. Viola and Jones have developed in [78] a real-time object detection system with automatic feature selection. Different solutions have

been proposed to deal with partial occlusions and cluttered backgrounds. Authors like Sali and Ullman [67] use a representation based on the appearance of the parts of the objects. Selinger and Nelson address in [68] the question of how much the performance of appearance-based methods can be improved in the presence of clutter and occlusion.

It seems clear that a combination of image features provides a better characterization of objects than a single feature on its own. In terms of object localization in image sequences, Mel combines color, shape and texture information in [49], and Papageorgiou et al. introduce motion information to segment people in video sequences in [59], yet they still follow the paradigm of static images. All these approaches to object detection in video disregard the temporal component. However, we already know that many objects show a characteristic dynamic behavior of their image features. Temporal textures like water or tree leaves have very particular motion patterns along time. The color of the blinking traffic light from chapter 3 changes from black to orange, and back. These behaviors can not be recognized in a single static image. Papageorgiou and Poggio [60] used dynamical information about the object by extending the static image approach through a set of consecutive frames. However, they do not exploit the 3-D space and time information structure of the video sequence. The previous chapter has shown that accumulating static image descriptors does not provide a proper representation of temporal processes.

In this section, we deal with the detection and localization of talking heads in news videos, which are relevant for indexing and annotation purposes. However, it is important to note that the method is not specifically designed as a face detector, but it provides a general framework for object detection in video.

4.1.1 CMC models for object detection and localization

For simplicity in the application of the CMC models for object detection and localization, in order to avoid additional processing of the image sequence, the experiments shown in this section are restricted to objects that only show local motions, and not global motions due to camera operation, for instance. Thanks to this constraint, we make sure that the model is only capturing dynamic characteristics produced by the object. Talking heads in news videos, and particularly news anchors, obey this constraint and will be the focus of this work. In the general case where global image transformations may exist in the sequence, the images should be registered and thus compensated for global motions.

The process of object localization is performed in three steps shown in fig. 4.1. First, the parameters of the model Ψ_0 that represents the target object are computed from a training sequence that only contains the object to be represented. Then, the test sequence is spatially divided into n rectangular blocks and the parameters of the temporal model are computed for each block, obtaining models Ψ_1, \dots, Ψ_n . Finally, the similarity between these models and Ψ_0 is computed using their symmetric likelihood ratio from eq. (3.16).

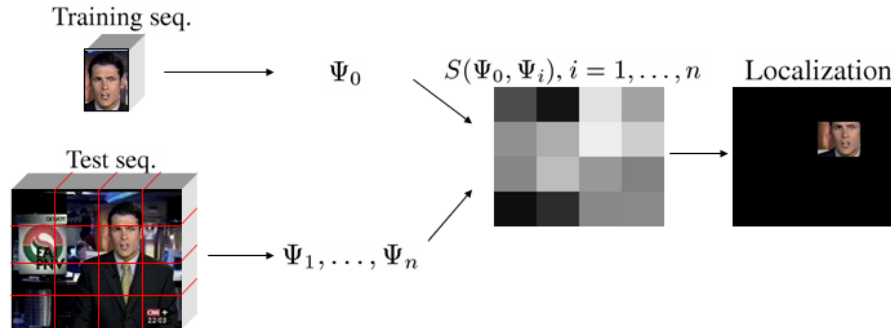


Figure 4.1: Object detection process in video sequences using temporal behavior models.

The choice of features depends on the application. In this case, talking heads have two main characteristics:

- the presence of a limited range of hues due to the color of the face,
- and a limited range of motions due to the distance to the camera.

The type of shot in terms of relative distance of the subject matter to the camera has been shown to be captured by motion descriptors in temporal models (see chapter 2). In our experiments, we will use the hue component of the HSV color space and the normal flow as scalar measures of the features, and will be handled as described in the previous chapter.

4.1.2 Experiments and discussion

The goal of our first test is to assess the need of a coupled color and motion model to describe the temporal behavior of the features instead of a simpler independent features model, or even single feature models. The training sequence (fig. 4.2(a)) was generated by spatially cropping a shot that was used as test sequence as well (fig. 4.2(b)). The test sequence was divided into 64 blocks. Figure 4.2(e) shows the similarity between the training sequence and each test block using the independent features model. We can observe the contribution of each individual feature to the model. Color (fig. 4.2(c)) is much more meaningful than motion (fig. 4.2(d)). The motion of the head is very subtle and is confused with the static background. The blocks of the test image where there is visible skin on the head have a significant similarity to the model. However, other parts of the scene also have high similarity caused by the black regions of the hair, background and jacket in the training sequence. Furthermore, skin color is likely to be found in objects other than heads.

When both features are coupled (fig 4.2(f)) the model becomes more robust. It captures subtle motions associated to skin and dark colors. The maximal similarities

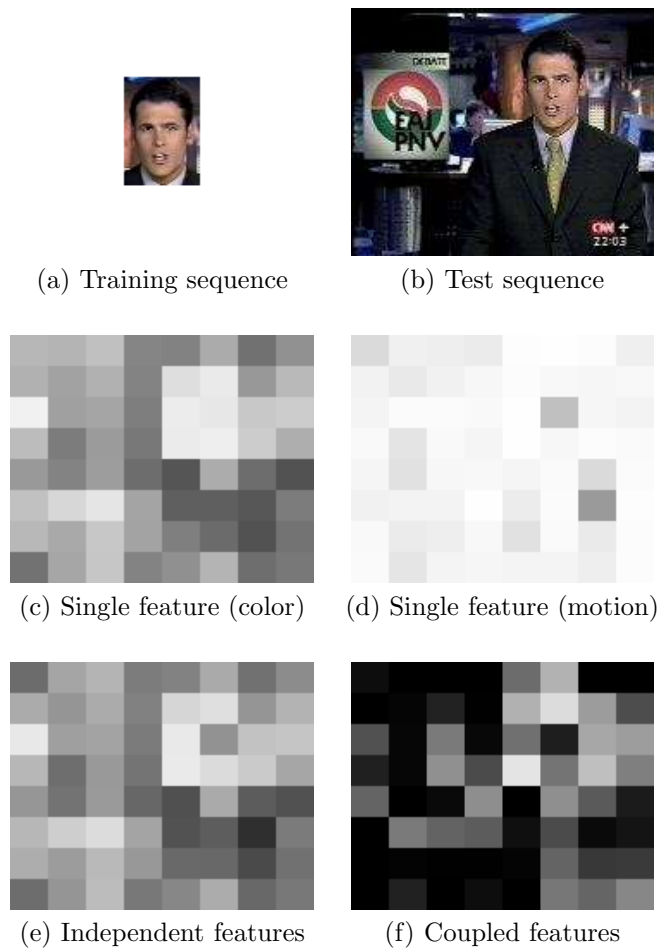


Figure 4.2: Similarity between the training sequence and the blocks of the test sequence using single feature, independent features, and coupled models. Higher similarities in (c) through (f) are encoded with whiter patches.

are found in the chin/mouth and forehead/hair blocks of the test sequence, where the normal flow is more evident and couples better with color information. Note that the dissimilarity between different regions is much more evident using this model. Many different objects can share similar colors or motion patterns. However, it is unlikely to find different objects that share the same color, motion, and dependency relationship between them.

The scale of the blocks of the test sequence is an important issue. A large scale (fewer blocks) provides a better estimation of the parameters of the model for each block. A small scale allows a better location of the object in the image, although the estimation of the parameters is less accurate and we may have only partial information about the object (e.g. only the chin and mouth instead of the whole head). Figure 4.3 shows the similarity between the training sequence and the blocks of the test sequence at different scales. The similarity is consistent from larger to smaller scales, which allows us to define a coarse-to-fine strategy to locate the object in the image with fewer computations. Furthermore, we observe that the similarity in the best block is significantly higher in the scales that produce blocks with the closest size to the object in the training sequence. In our example, the most suitable scale is between 2×2 and 4×4 . On the other hand, small scales are useful to know what parts of the object are the most representative in the model. In this case, these parts correspond to chin, mouth and hair.

Figure 4.4 shows examples of detection and location of the talking head of the anchorman under different variations of scale, position and color acquisition conditions. The first test sequence shows a special case where the head is always between at least two blocks. The symmetry of the head makes both blocks have a high similarity, although they only contain a half of the object. In the second one, a change in scale affects the motion measure. The true motion becomes smaller in the image when the object gets farther from the camera. This sequence was also digitized using different color acquisition parameters that cause a variation in the skin color of the face.

Although a certain variation is allowed, the model trained using a single sequence does not generalize to other objects of the same class, that is, to other talking heads. We can train the model with more than one sequence by accumulating their cooccurrence matrices before parameter estimation. In this way, larger variations are allowed in test sequences, which also mean that non-object blocks get more similar to the training sequences as well. When the model is trained using several sequences with significant variations, a wider range of bins in the cumulative cooccurrence matrix become non-zero. Therefore, there is a slightly higher chance of confusion between positive and negative examples. Figure 4.5 shows that the average similarities of the best block in positive and negative examples get closer when the model accepts larger variations of the object (the scale was fixed at 4×4). The discriminability between positives and negatives is higher with the specific model.

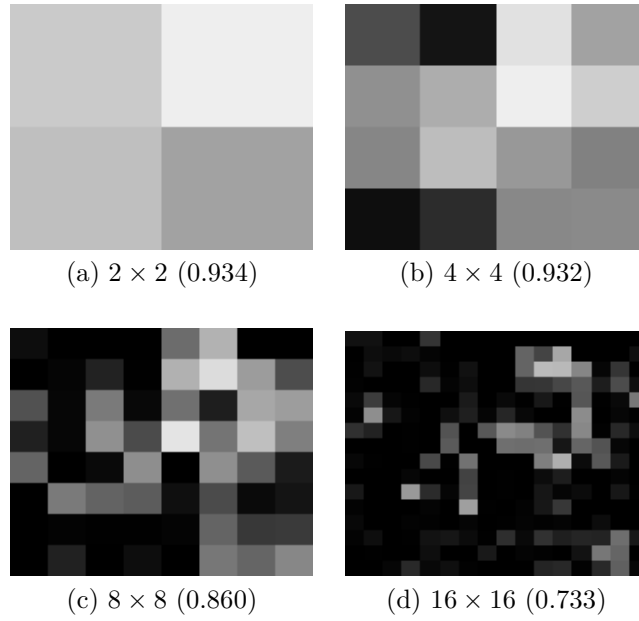


Figure 4.3: Similarity between the training sequence and different sizes of the blocks of the test sequence using the coupled model. The similarity in the best block is shown in brackets. The best similarity (whitest patch) is given at the most suitable scale. The training and test sequences are the same as in fig. 4.2.

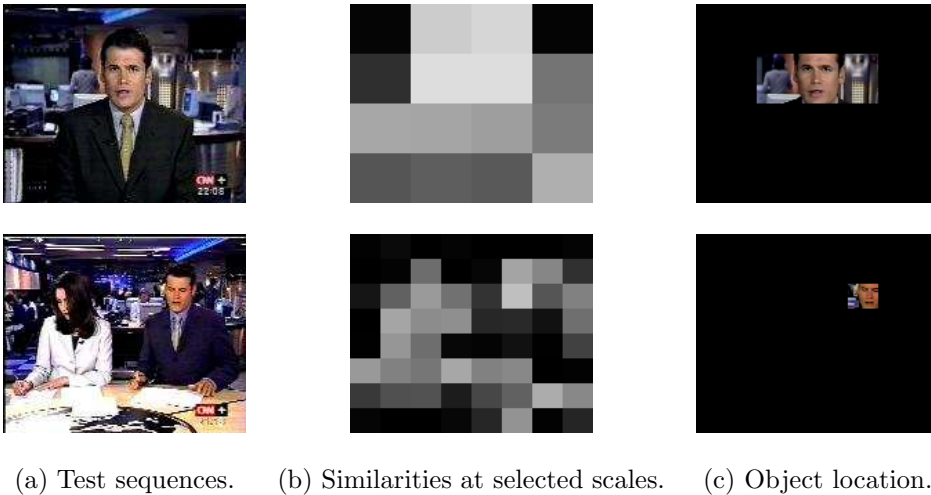


Figure 4.4: Detection and location of the anchorman with variations in scale and color acquisition conditions.

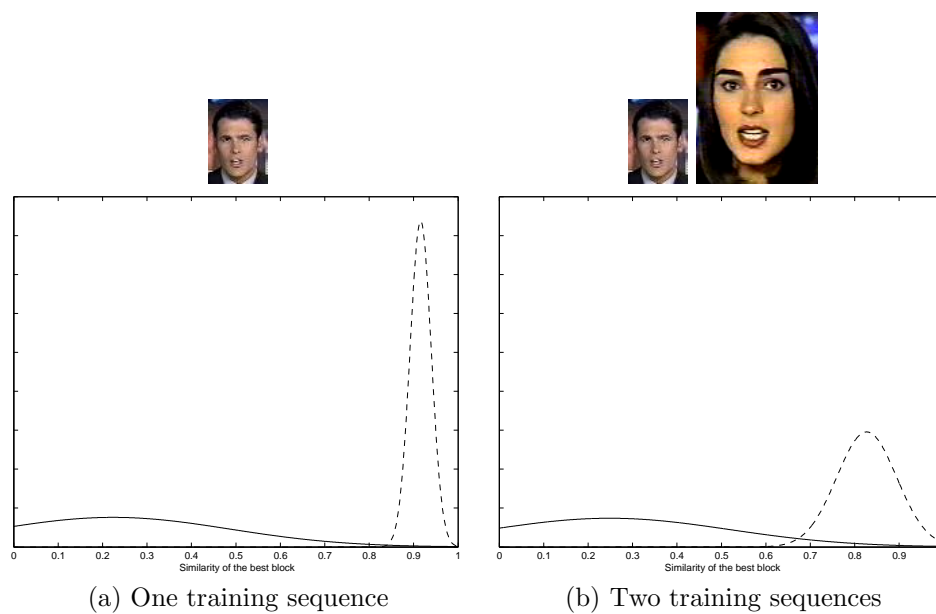


Figure 4.5: Probability distribution of the similarity measure in the best blocks of positive (dashed) and negative (solid) examples using (a) one sequence for training, and (b) two sequences in order to consider more variations of the object. The two training sequences have different sizes and color saturation due to different acquisition conditions. The similarity of the best block in the test sequences to the training sequence(s) is in the x axis. The similarity between the two training sequences was 0.798.

4.2 Shot boundary detection

Shot boundary detection is the basic first step for indexing and organizing digital video assets. In the review of prior work on shot boundary detection from chapter 1, we have identified two main issues of commonly used approaches:

- The selection of a fixed predefined threshold is difficult, and it usually does not exist.
- The frame-to-frame comparison approach works well for sharp cuts, but it is not appropriate for gradual transitions.

During the process of obtaining the CMC-based representation of video shots, the same procedure can be used to obtain shot boundaries. This method has several advantages. It allows us to combine multiple features in the same representation. Also, information from all the frames since the beginning of the shot is kept in the representation, instead of using a simple frame-to-frame comparison. Finally, an adaptive threshold that only depends on the distance measures obtained during the process is defined, so that a fixed threshold is avoided. The CMC-based shot boundary detector can also be used as a stand-alone process, in case the representation of shot contents is not needed.

4.2.1 Using the CMC representation to detect shot boundaries

A change of shot is characterized by a change of the contents in the images (either sudden or gradual). Given the representation of visual contents in an image sequence briefly discussed above, we can define a shot segmentation scheme that checks the consistency of the transition into a new image with respect to the images already contained in the representation. That is, we can compute how well the observations attached to the next step in the image sequence fit a probability distribution obtained from the previous images. This can be expressed as:

$$D_{t+1} = P(I_t \rightarrow I_{t+1} | I_1 \rightarrow I_2 \rightarrow I_3 \cdots I_{t-2} \rightarrow I_{t-1} \rightarrow I_t) \quad (4.1)$$

where $I_i \rightarrow I_j$ represents image feature transitions between images i and j of the sequence. If the transition from I_t to I_{t+1} fits the probability distribution, then it is included in the representation. Otherwise, a shot change is detected. The KLD measure from eq. (3.17) can be used by defining the observations from the image sequences $x^{\Psi_1} = \{I_t, I_{t+1}\}$ and $x^{\Psi_2} = \{I_1, \dots, I_t\}$, and computing the parameters Ψ_1 and Ψ_2 of their corresponding distributions. Using KLD has the advantage that more images can be considered in the sequence to be compared in order to obtain better estimates of the parameters. In this way, the observation x^{Ψ_1} can include not only the transitions $I_t \rightarrow I_{t+1}$, but also the transitions $I_{t+1} \rightarrow I_{t+2}$, and so on.

The main advantages of the CMC-based approach with respect to most shot segmentation algorithms found in the literature are:

1. It is not based on the degree of correlation between adjacent frames. The contents of all the images from the beginning of the shot are considered in the representation.
2. Multiple features can easily be integrated in the representation in order to obtain a more robust detection.

Besides all the disadvantages that can be enumerated when a fixed pre-defined threshold is used, the selection of a detection threshold is particularly difficult in this case. The probability distribution that represents the images in the shot gets more accurate as the number of images considered grows. When the number of observations is large, a better estimation of the parameters is obtained. At the beginning of the shot, we may have a less accurate estimation and the distances computed can be higher than when the estimation is correct. For this reason, a fixed threshold can not be used in order to detect shot boundaries and we have defined an adaptive threshold. If we compute the mean μ and standard deviation σ of the distribution of distance measures from the beginning of the shot, μ will tend to a value that depends on the contents, and σ will tend to 0, as the distribution representing video contents gets more accurate. The adaptive threshold can be established, for instance, at $thr = \mu + 3\sigma$, so that distance values that do not correspond to expected values will be detected. Note that this threshold only depends on the contents, and that no model is defined on, for example, shot duration like in other approaches found in the literature [76].

4.2.2 Experimental results

Experiments have been focused on a short sequence of 2000 frames from a news video. This sequence was particularly selected in order to analyze two main things:

1. the selection of a fixed pre-defined detection threshold vs. the use of an adaptive one,
2. the improvement achieved by coupling multiple image features in the model with respect to the use of individual features alone.

The interest of this test sequence is found in the variety of transition effects in it: 8 cuts and 5 gradual transitions (4 wipes and 1 dissolve). The location and type of these transitions are detailed in table 4.1. Besides, there are two complex computer-generated sequences that mark the beginning and the end of the news summary (see fig. 4.8(a)).

The image features considered in these experiments were color and motion. Many shot segmentation methods have been based on these two features. As always along this thesis, the color feature is the hue component from the HSV color model, while the motion feature is the normal flow. Each feature is computed for every non-overlapping 16×16 image block. In this case, both features were quantized in 8 levels.

Frame number	Transition
246	Cut
374	Cut
568	Wipe
671	Cut
727	Cut
767	Wipe
850	Cut
938	Cut
964	Wipe
1154	Cut
1187	Wipe
1405	Dissolve
1527	Cut

Table 4.1: Location and type of the shot transitions in our test sequence.

The problems of a fixed pre-defined detection threshold are shown in fig. 4.6. The plots show the distance measure defined combining eqs. (3.17) and (4.1) as a solid line, and the threshold value as a dashed line. All plots in this test were obtained using the coupled motion and color model. When the threshold selected is too high ($thr = 4$ in fig. 4.6(a)), false positive detections are avoided, but some actual transitions are missed. Particularly, wipes around frames 568, 767 and 964 were not detected. Moreover, the cut at frame 938 was not detected either. On the other hand, when the threshold is too low ($thr = 2$ in fig. 4.6(b)), gradual transitions can be correctly detected, but we obtain 20 false positive detections. Furthermore, the cut at frame 938 is still not detected. This means that the threshold should be even lower, and more false positive detections would be reported. We can conclude that a fixed threshold is very difficult to define, and in many cases there will not exist an appropriate threshold. The results obtained with the adaptive threshold are shown in fig. 4.7. All the transitions in the sequence were correctly reported, with only 3 false positive detections at frames 1158, 1191 and 1482. Note that the probability distribution representing shot contents is initialized every time a shot boundary is detected. For this reason, a fixed threshold may report several detections during a gradual transition. The adaptive threshold minimizes these false detections because it depends on the distance measures, so that the threshold is high when distances are high too.

One of the computer-generated shots in the sequence spans from frame 1 to 245 (1 of every 40 frames are shown in fig. 4.8(a)). Figures 4.8(b) and (c) show detection results using single-feature motion and color models respectively. These plots are very noisy, especially with color. Several false positive detections are reported during the shot (2 with motion, 3 with color). On the other hand, when both features are coupled (fig. 4.8(d)), the plot is much smoother and no false positives are reported. Errors caused by one feature are compensated by the other one. Both features thus

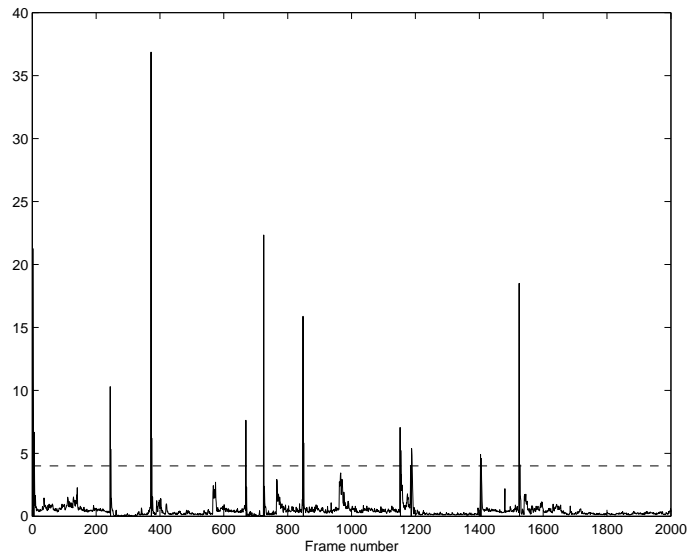
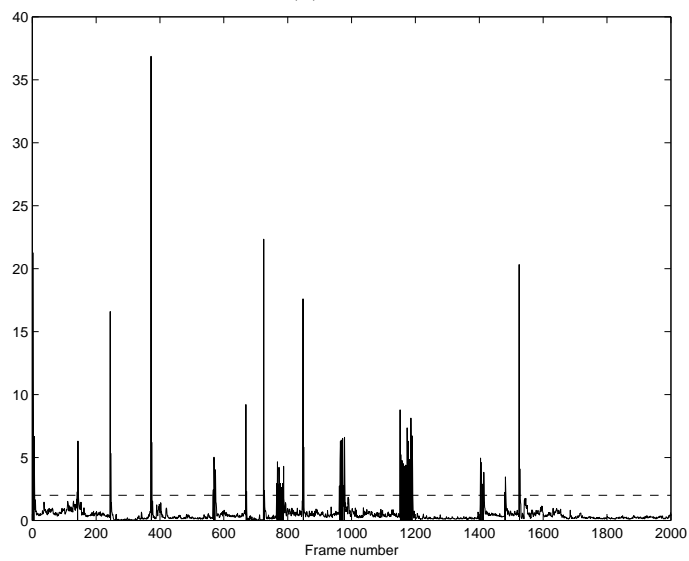
(a) $thr = 4$ (b) $thr = 2$

Figure 4.6: Selection of a fixed detection threshold. A high threshold (a) misses some boundaries, while a low one (b) reports too many false detections. The threshold is shown as a dashed line.

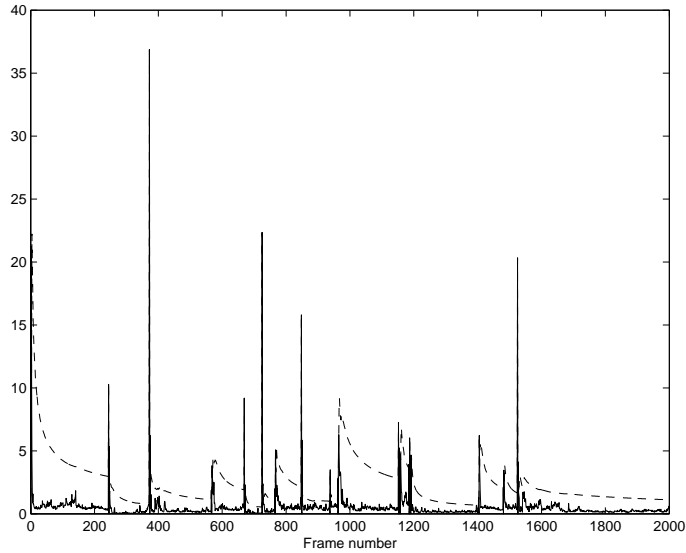


Figure 4.7: Shot segmentation results using the coupled model of motion and color features and the adaptive threshold, shown as a dashed line.

Feature	Correct	Missed	False	Precision	Recall
Motion	11	2	22	.33	.85
Color	13	0	21	.38	1
Coupled	13	0	3	.81	1

Table 4.2: Summary of results using single-feature and multiple-feature models on our short test sequence.

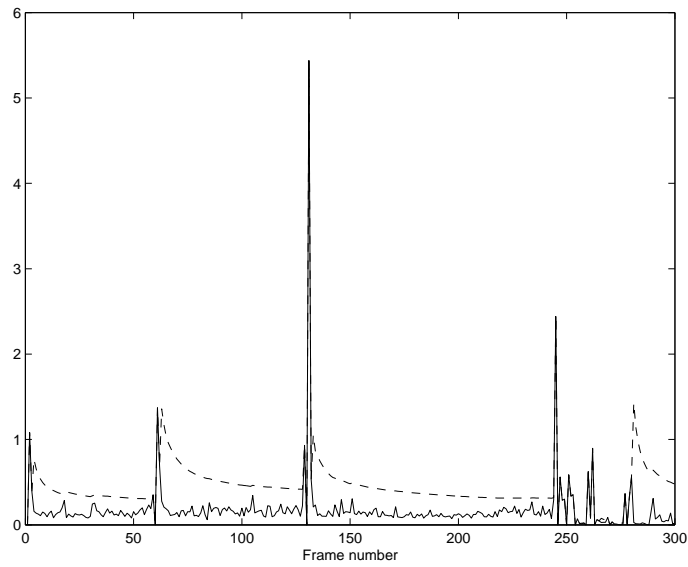
cooperate in order to better determine when a real shot boundary is found and the behavior of both color and motion change, and also when we are still in the same shot and one of the features may have changed but the other keeps the same behavior. Considering the full video sequence, the results summarized in table 4.2 are obtained. Single-feature models have good recall, i.e. most actual transitions are correctly detected. However, they are quite unstable in the sense that the variations in the distance measures are too significant and many false positive detections are reported. That is, their precision is low. The combination of multiple features in the model shows higher precision. In other words, the detection is more robust and less noisy.

4.2.3 Summary of shot boundary detection using CMC

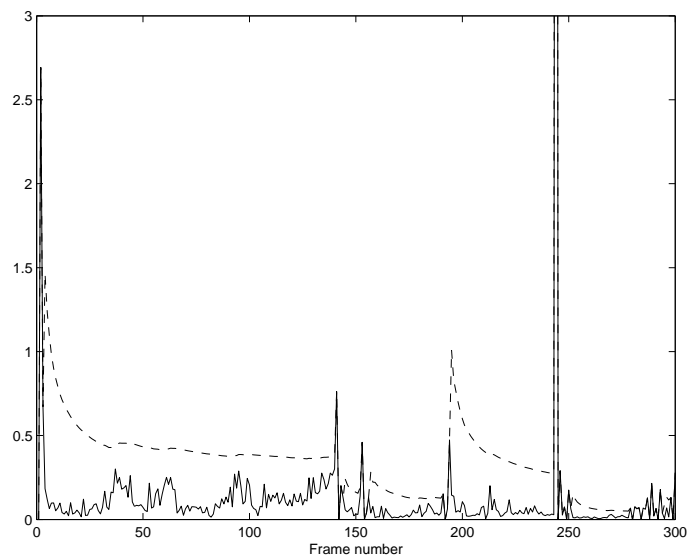
The CMC-based method for shot boundary detection has two main advantages over other algorithms specifically designed for this purpose:



(a) Frames from a complex computer-generated shot.

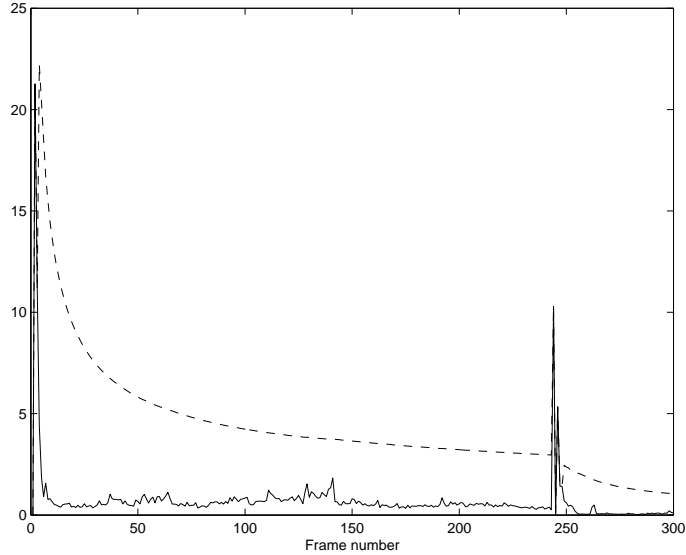


(b) Motion feature.



(c) Color feature.

Figure 4.8: (a) Frames from a complex computer-generated sequence (frames 1-245 of our test sequence). (b) Motion and (c) color features individually do a poor job and report false positives. (d) When they are coupled, one compensates the errors of the other. The adaptive threshold is shown as a dashed line.



(d) Coupled features.

Figure 4.8: (Continued).

1. Multiple image features can be easily combined in the same representation, thus providing a more robust detection of shot boundaries.
2. Information from all the images since the beginning of the shot is kept in the representation, so that not only adjacent frames are compared.

Using a particularly selected test sequence that contains a variety of shot transitions and complex computer-generated shots, experimental results lead us to the following conclusions:

- The selection of a fixed pre-defined detection threshold is usually difficult, and many times it is not appropriate for the video contents subject to analysis.
- An adaptive threshold that depends on the distance values that are computed is more appropriate in order to allow the method to work correctly on different video contents.
- The combination of different image features in the same model provides a more robust representation than each of them individually. Color and motion features cooperate in order to better detect actual shot boundaries and avoid false detections.
- Both abrupt and gradual transitions are detected by this method with high recall and precision.

4.3 Intermediate-level semantic clustering of shots

This section analyzes how the CMC representation of shot contents can extract complex characteristics of the scene using very simple image features. The experiments detailed next are based on the analysis of unsupervised clusterings of the shots of a video. Considering the low-level image features used in the representation, hypotheses on how the shots should cluster together can be formulated, and compared to the results obtained. Experiments will be conducted on two different video domains:

- Sports: The Soccer sequence is 1,100 frames long, and has 10 shots.
- News: The News sequence contains 73 shots and over 24,000 frames.

After the review and analysis from chapter 2 about intermediate-level semantics from low-level features, color and motion will be used as reference low-level image features, due to the amount of semantics they convey. Again, color will be represented by the Hue component from the HSV color space and motion by the normal flow.

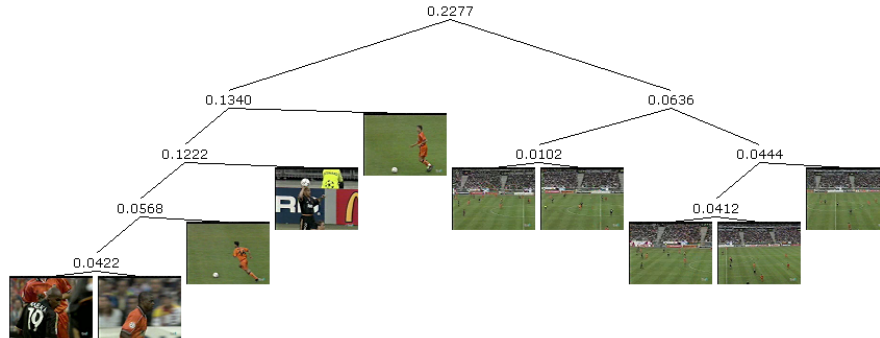
The symmetric KLD dissimilarity measure from eq. (3.18) will be used to build the clusters. The initial clusters contain single shots and they are successively merged following a minimum distance (or dissimilarity) criterion. The distance between two clusters \mathcal{C}_∞ and \mathcal{C}_ϵ is given by:

$$D(\mathcal{C}_1, \mathcal{C}_2) = \max\{S_{KLD}(\Psi_i, \Psi_j), \forall \Psi_i \in \mathcal{C}_1, \Psi_j \in \mathcal{C}_2\} \quad (4.2)$$

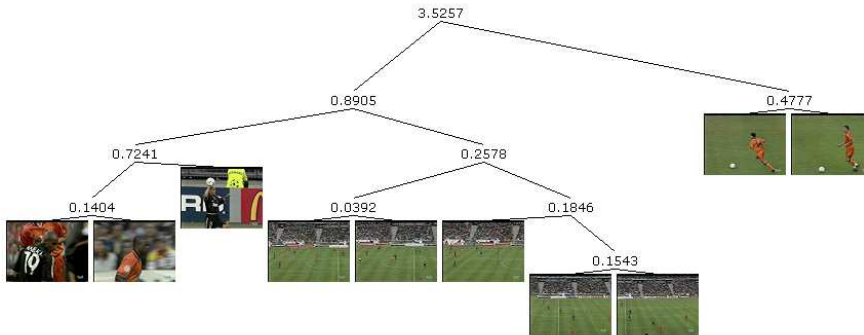
The result of this hierarchical clustering is naturally represented by a binary tree, where the leaves are associated to shots, and inner nodes contain the distance between the two clusters it joins, and possibly other useful measures.

4.3.1 Analysis of the Soccer sequence

In this video sequence, a model that only considers color is useless in practice. Most of the shots show a large green surface, so that we would basically characterize whether the target matter is on the playing field or not. Motion models provide more information. A model of motion of the shots basically characterizes the type of shot, i.e. close-up, medium, long, and so on. In this way, different types of shots are clustered together. Mainly, global views of the plays vs. medium shots of the players in different situations of the game are characterized (see fig. 4.9(a)). The combination of color and motion information adds valuable information in order to accurately represent these situations, as shown in fig. 4.9(b). In this case, three different classes can be differentiated: wide angle shots that follow game play, player close-ups and player-action shots. This semantic information can be used to obtain higher level structures of the game, to annotate game plays, or to trigger other processes like face recognition to know the identity of players in a close-up, or even jersey number recognition.



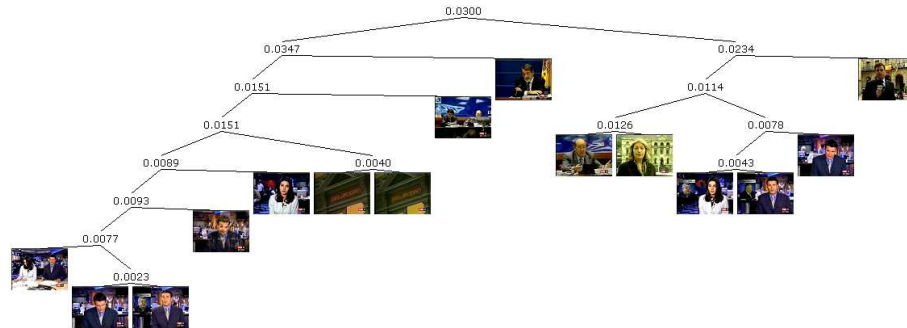
(a) Clustering using motion MC model. The clusters are based on the type of shot. Wide angle and player shots are differentiated.



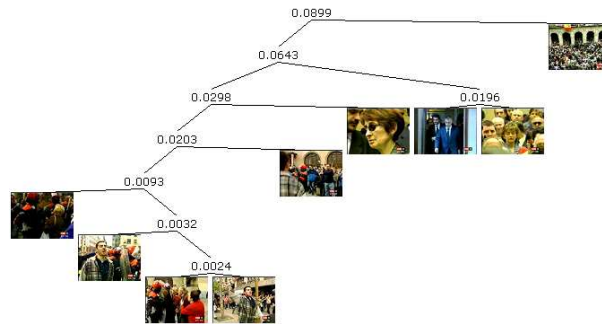
(b) Clustering using motion and color CMC models. We can differentiate three classes: wide angle shots, player close-ups and player-action shots.

Figure 4.9: Clustering of the Soccer sequence using different models and the level of visual contents they can characterize.

Note that a domain-dependent pre-processing of the images could be applied depending on what kind of information we want to characterize. For instance, camera operation can be considered as a source of noise for the motion observations. Assuming that the dominant motion is caused by the camera, we may want to compensate the images for global motion. In this way, we would expect to obtain a better characterization of the activity of the objects in the scenes. However, it may be interesting to include the camera operation component in the model, depending on the domain and its application. In the Soccer sequence, one may also consider that the dominant green color of the field is a source of noise for the color observation. It is easy to isolate green pixels and disregard their corresponding observations.



(a) Cluster of “talking heads”.



(b) Cluster of “crowds”.

Figure 4.10: Clusters from the News sequence using only motion information.

4.3.2 Analysis of the News sequence

In this sequence, we can also expect a model of motion of the shots to characterize the type of shot. A motion model also captures any motion particularities that the objects in the scene may have. For instance, recalling chapter 2, a talking head has a motion pattern characterized by a dominant subtle motion with sudden fast movements, both in random directions. Figure 4.10 shows two interesting clusters obtained using only motion information. The first one is a cluster of “talking heads”, while the second one has mainly grouped shots of “crowds”. Also, note that cluster distances in these two clusters are small, which means that the clusters are compact and thus the semantic concept is well defined. Therefore, although motion is basically associated to the type of shot, the higher capacity for representing contents of the CMC models, in front of the simple temporal motion cooccurrences used in chapter 2, can capture finer structure in the temporal motion patterns that leads to a better characterization of more complex concepts like “talking heads”, “crowds” or “sports” only using motion information.

On the other hand, color provides very significant information about location of the scene, e.g. a studio, a street outdoors or a conference room. Anchor shots have

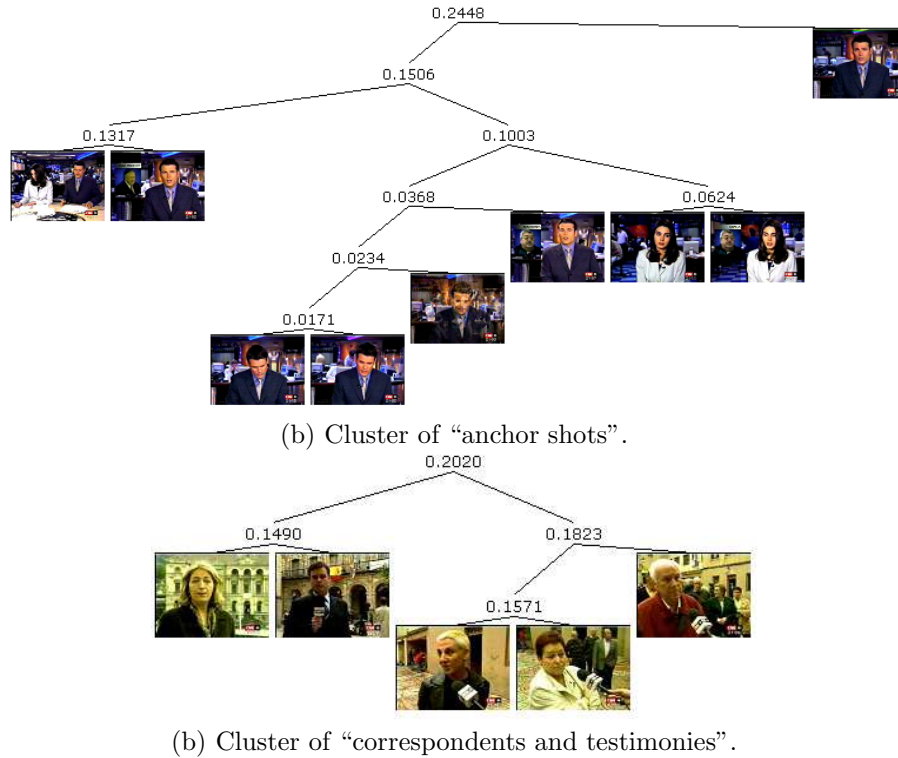


Figure 4.11: Shots clustered by both activity (motion) and location (color) using the CMC model.

very particular colors due to their studio location, so that the single feature model is enough to characterize them. The CMC model couples the contribution of both features. New semantic classes emerge by the combination of the motion activity and the location components. The more generic class of “talking heads” is partitioned by location, so we can differentiate between “studio anchor shots” and a new category that includes “correspondents and testimonies” within the same news segment and location. That is, motion gives the clue to know that a talking head is in the scene and color provides information about the location. These two clusters are shown in fig. 4.11. Moreover, the combination of both features adds information about how the skin color of a head moves when the head is talking, so that visual contents are more accurately represented.

Other semantically interesting clusters can be found in the tree obtained from the News sequence. For example, the cluster shown in fig. 4.12 has grouped shots from the Sports section of the news program, which contain the reactions of soccer players when they score a goal. These clusters show the ability of the CMC representation to capture the semantics of contents.

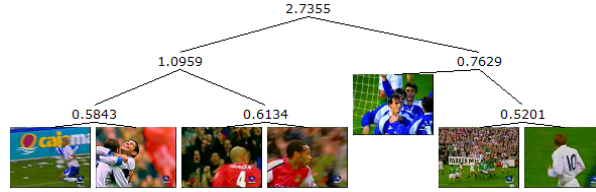


Figure 4.12: Cluster of soccer player reactions when they score a goal, found in the Sports section of the News sequence.

4.4 High-level structuring of news videos

In chapter 1, news videos were shown as an example of how the high-level semantic structure of a video can be automatically obtained using intermediate-level semantics and domain knowledge. An analysis of the news domain led us to two main observations about its typical production model:

- News items begin with an anchor shot.
- Footage from news items is re-used in the summary at the beginning of the program.

These facts allow us to automatically obtain the structure shown in fig. 1.1, once information about anchor shots and shot similarities is available. The identification of anchor shots will allow us to segment news items. Then, the footage from the summary found at the beginning of the video can be matched to the shots in these already segmented news items. Temporal information is inherent to the shots and is obtained during shot segmentation, so that it will be available through all the process.

The hierarchical intermediate-level semantic clusterings from the previous section contain enough information to obtain the structure of a news video using the previous rules. Anchor shots are very characteristic in news videos. They appear a relatively high number of times (compared to the rest of the shots in the newscast), and their image features are always very similar, like the color of the background (even in the case of different newscasters) and the characteristic motion of a talking head. Therefore, we can expect anchor shots to form a compact cluster with a relatively large number of elements in it. The compactness of a cluster is inversely proportional to the maximum distance between its elements. We thus define the following measure:

$$d(\mathcal{C}_i) = \frac{|\mathcal{C}_i|}{\max\{S_{KLD}(\Psi_i, \Psi_j), \forall \Psi_i, \Psi_j \in \mathcal{C}_i\}} \quad (4.3)$$

which is the density of the cluster \mathcal{C}_i . Note that the maximum distance is already computed during the clustering process. A compact cluster with a large number of elements will have a high density.

The same clustering will be useful to associate shots from the summary (ToC) with shots from their corresponding news items. If the shots come from the same

video footage, they will be almost identical, or at least very similar (maybe with different lengths, for example). Therefore, they will be joined in very early stages of the clustering process, so that we can look for shots with the following characteristics:

1. very low distance between them,
2. one of the shots is found at the beginning of the temporal sequence of shots in the news video, and the other one belongs to a news item (previously identified using anchor shots).

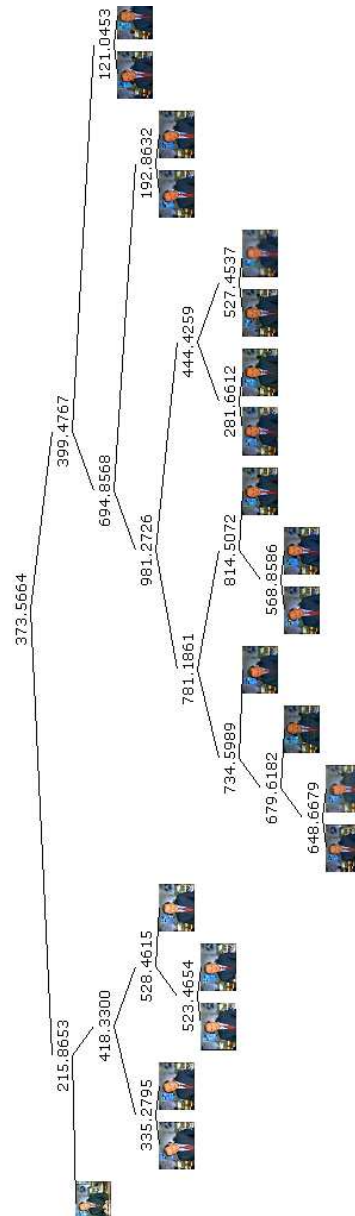
4.4.1 Results and discussion

We have selected a full news program from a Spanish station to evaluate our proposal. The video is 50,119 frames long (33 mins. and 24 secs.), and has 651 shots. It follows the typical structure of a summary at the beginning and news items starting with anchor shots.

Once the shots have been segmented and represented using the CMC model that combines color and motion, the semantic clustering was obtained as in the previous section. Cluster densities were then computed for all the nodes of the tree using eq. (4.3). Figure 4.13 shows different clusters obtained in this tree. In this case, inner nodes show the density of the clusters formed by the elements under them, instead of the distance between the two clusters joined at that point. This particular news program has two different anchors, one for Sports news, and one for the rest. Their shots are clustered together in the two clusters shown in figs. 4.13 (a) and (b). The densities of these clusters (and also of their subclusters) are high compared to other clusters that group together more disparate shots, like in fig. 4.13 (c). The average cluster density through the whole tree is 62.52. It is important to note that in this tree only 11 nodes have more than 3 shots under it and a density over 200, and only 4 nodes have more than 10 shots and density over 200. These 4 nodes are subclusters in the cluster of main anchor shots (fig. 4.13 (a)).

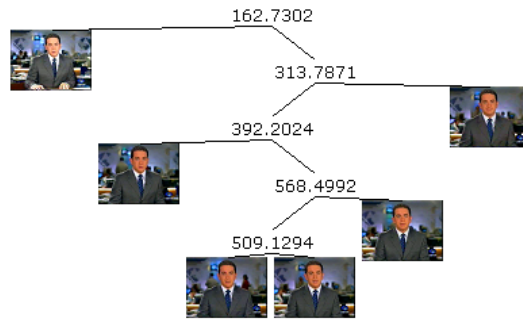
The clustering also allows us to link elements from the ToC to their corresponding news items. Figure 4.14 shows shots from the ToC and their most similar shots in the whole video. We know that the shots in the left belong to the ToC because their timestamps are from the beginning of the video, and the shots in the middle belong to news items because they have been already segmented. On the right side, we show the parts of the clustering tree where these shots are joined, and their symmetric KLD measure. For a better notion of the goodness of these distances, note that the average symmetric KLD through the whole tree is 0.74. In this case, semantic information is not necessary, as only visual similarity between shots is considered. The clusterings obtained using CMC models thus establish two kinds of similarity:

- Visual similarity, based on raw similarities of low-level image features.
- Semantic similarity, based on the semantics conveyed by low-level features, which combined characterize semantic concepts.

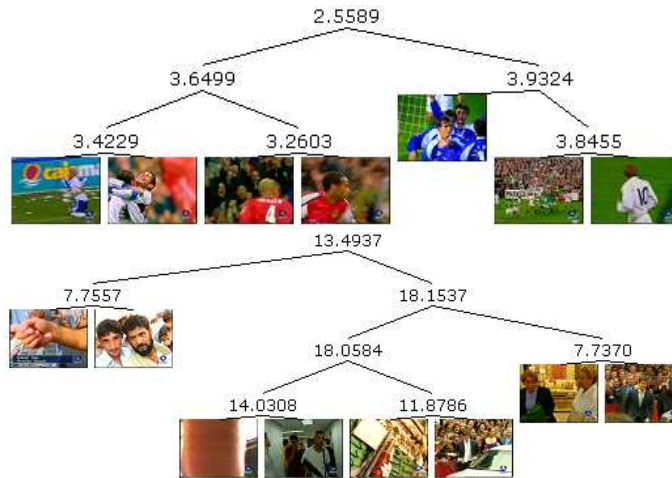


(a) Cluster of main anchor shots.

Figure 4.13: Some clusters from a news video. Inner nodes show the density of the clusters (and subclusters). The average cluster density in the full tree is 62.52.



(b) Cluster of sports anchor shots.



(c) Arbitrary clusters.

Figure 4.13: (Continued).

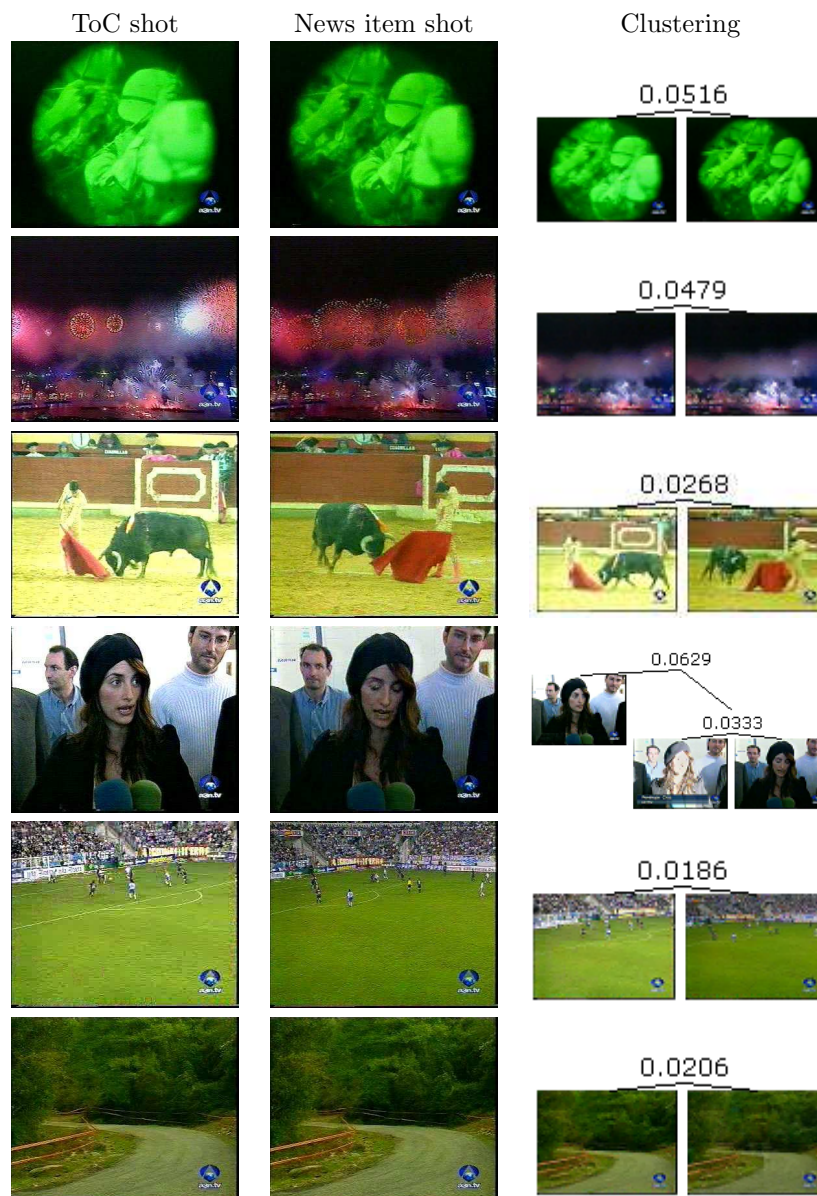


Figure 4.14: Some shots from the ToC (left) also used in their corresponding news items (middle). They can be located in the clustering tree with a very low symmetric KLD (right). The average symmetric KLD in the full tree is 0.74.

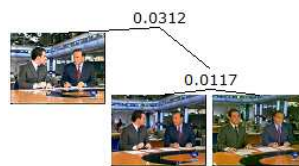
We have used very simple rules for extracting the structure of a news video. The first rule may fail when the newscaster interviews another person. In this case, the alternating shot structure of an interview can be detected with the same methods used so far. The shots of the interviewee would form a separate compact cluster in the tree. A set of interlaced shots from two different compact clusters would represent an interview. There are several other clusters that can help to better identify and automatically annotate the structure of news videos. Some examples are shown in fig. 4.15:

- Global studio shots provide information about the change of section. The case from fig. 4.15(a) shows both the main and the sports newscasters. In this way, the Sports section can be identified.
- Dual connections are often established with foreign correspondents in order to interact with them. In these cases, the screen is partitioned into two regions, one for the local newscaster and one for the correspondent, like the frames shown in fig. 4.15(b).
- Shots of the same or different events of the same sport cluster together. In Spain, most of the time for Sports news is devoted to soccer, as shown in fig. 4.15(c). Characterizing soccer shots can also lead to the automatic identification of the Sports section. In other countries, baseball, basketball or cricket would do the same job.

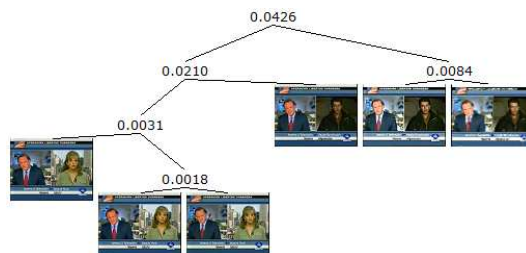
4.5 Summary

This chapter has shown the ability of the CMC modeling of video contents to capture intermediate-level semantics from very simple low-level color and motion features. As a generalization and extension of histogramming techniques, the CMC model can be used for object detection and localization in video, which is made much more robust by combining multiple features and taking advantage of their dynamical behavior. In the process of obtaining the high level semantic structure of videos, the CMC representation offers an intermediate-level semantically meaningful clustering of shots, which allows us to automatically extract higher level structures using very simple rules about the domain. Its application to the domain of News videos has been proved successful, and with many more possibilities than the ones exploited here. A robust shot boundary detector appears in the same process of computing model parameters for the shots of the video. In this way, the whole process is unified and there is no need to apply a shot segmentation algorithm prior to the representation of their contents. The CMC representation using color and motion provides an intermediate-level semantic description of contents, given that it contains information about:

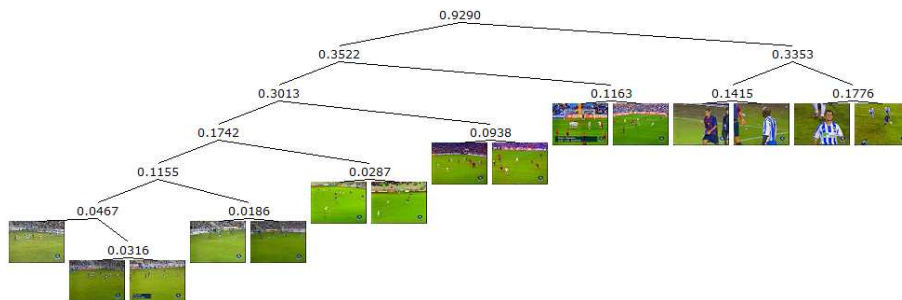
- Objects, their identity, relative size and motion.
- Camera operation given by global motion.



(a) Global studio shots.



(b) Dual connections with correspondents.



(c) Footage from sports news, mainly soccer in Spain.

Figure 4.15: Clusters that may help to better identify and automatically annotate the structure of news videos.

- Temporal relationships between elements of the scene.
- Type of shot provided by motion information.
- Location provided by background colors.
- Global sense of activity.

Besides, other intermediate-level semantic concepts like “crowds” or “talking heads”, which can further be classified into “anchor shots” and “correspondents and interviewees”, are also characterized by this model and can be very helpful for automatic annotation of videos, particularly in the domain of News.