INFORMATION-THEORETIC FRAMEWORK FOR THE JOINT TEMPORAL PARTITIONING AND REPRESENTATION OF VIDEO DATA

Bruno Janvier, Eric Bruno, Stéphane Marchand-Maillet, Thierry Pun

Viper Group, Computer Vision and Multimedia Laboratory
Université de Genève
janvier@cui.unige.ch

ABSTRACT

The first step in the analysis of video content is the partitioning of a long video sequence into short homogeneous temporal segments. The homogeneity property ensures that the segments are taken by a single camera and represent a continuous action in time and space. These segments will then be used as atomic temporal components for higher level analysis like browsing, classification, indexing and retrieval.

The novelty of our approach is to use color information to cut down the video into segments dynamically homogeneous using a criterion inspired by compact coding theory. First, we use a statistical detection framework to detect abrupt "shot" transitions (strong discontinuities in the data stream), then, we perform an information-based segmentation inside each "shot" using a Minimum Message Length (MML) criterion and minimization by a Dynamic Programming Algorithm (DPA).

We show that our method is robust to detect all types of transitions in a generic manner. A specific detector for each type of transition of interest becomes unnecessary. We give two examples of applications : shot boundaries detection and keyframe selection.

1. INTRODUCTION

The increasing amount of video documents produced every day creates a new need for the management and retrieval in multimedia information systems. The first goal to achieve in this area of research is the temporal partitioning of any video in sub-sequences representing a continuous action in time and space for the purpose of further indexing.

The problem of shot-boundary detection has been tackled by many computer vision scientists without being completely solved. In the survey of Koprinska and Carrato [1], a number of different techniques of temporal segmentation of uncompressed or compressed video are described. Many methods are related to the detection of discontinuities using pair-wise pixels, block based or histograms comparisons. In the compressed domain, DCT coefficients are used instead of pixel values.

If the detection of a discontinous camera cut (hardcut) in a video sequence is relatively easy, a transition can also be due to a special-effect transition like a dissolve, a fade or a wipe. A transition in the action can also be due to the fact that the camera shows one thing at a given time and shows a completely different thing at another time; the transition can simply be a rotation or a zoom of the camera. To detect these events is also important for indexing purposes. There are plenty of different types of transitions that do not show any abrupt discontinuities (due to the presence of special effects or not) and their detection is therefore difficult as shown in the survey from Kasturi and al. [2]. It is proposed in many articles to design a specific detector for different transitions [3], [4]. Here, we rather depart from this solution in order to avoid ad-hoc techniques.

The novelty of our approach is to use color information to cut down the video into dynamically homogeneous segments using a criterion inspired by compact coding theory. We achieve this by means of two successives steps. First, we partition the video into continuous segments using an hardcut detection based on statistical hypothesis testing. Then, an information-based segmentation using a Minimum Message Length (MML) criterion will be applied inside each continuous segment to take into account all the available information and partition the video into segments where the evolution is homogeneous. In the literature, many clustering-based segmentation methods exist that use, for example, hierarchical clustering of frame dissimilarity. The approach we have chosen is different, because the number of segments and the location of the boundaries are infered in order to maximize the homogeneity of the evolution within each segment using a Dynamic Programming algorithm. This optimization process is global and therefore more satisfying than greedy or agglomerative strategies. As a result, the boundaries are located accurately.

This framework enables us to detect all types of transitions of the video and provides atomic temporal components for higher level content-based video analysis. Many

problems are simplified. It seems easier to classify these segments to find out if they represent a transition or an interesting content rather than to detect and recognize every possible transition. The keyframe selection problem is reduced to the problem of choosing the most representative of each homogeneous segment; similarly the shot boundaries detection problem is reduced to a post-processing to merge appropriate segments.

2. COLOR DISSIMILARITY PROFILE

In order to perform a temporal segmentation of the video stream, we need a distance measure between two successive frames. The analysis will be done on the resulting temporal profile of the frame-by-frame distances.

At the very least, the metric between two frames should satisfy the following properties :

- it should be stable with respect to changes that are common during a segment representing a continuous action in time and space such as small euclidian transformations, lighting changes, small Euclidean deformations, appearance of objects, etc.
- it should give an accurate quantitative information about the amount of change that has taken place.

The color histogram has proven to be a very stable representation in the content-based image retrieval research field. The distribution of color is invariant and stable for frames representing a similar content. We will compute the histogram in the opponent color space, because it is said to give the best performance/speed ratio in the comparison from Kasturi and al. [5].

The Jeffrey divergence is used to compute the distance between the histograms two by two. It represents how compactly one histogram can be coded using the other as a codebook and gives better quantitative results in our experiments than the L_1, L_2 or chi-square metrics. If H_i and H_j represent two histograms containing N bins, the Jeffrey divergence is defined by :

$$D(H_i, H_j) = \sum_{k=1}^{N} [H_i(k) log \left(\frac{H_i(k)}{m(k)} \right) + H_j(k) log \left(\frac{H_j(k)}{m(k)} \right)] \quad (1)$$

where $m(k) = \frac{H_i(k)+H_j(k)}{2}$.

We have now a frame-by-frame dissimilarity profile of the video. A discontinuity in the dissimilarity profile will appear as a strong peak and can therefore be detected quite efficiently.

Note that, in our framework, video information is abstracted by its features. In this respect, it is possible to replace color information, for example by motion or sound, and get a partitioning that will hold a different interpretation than that obtained with the methods that will be presented next.

3. DETECTION OF DISCONTINUITIES

The hardcut detection is a binary-hypothesis test problem. We introduce two hypotheses :

- Hypothesis S : there is a boundary present between frames k and $k+1$
- Hypothesis $\bar{S}$: there is no boundary present between frames k and $k+1$

The test can fail if we make a false detection (i.e. S is chosen when $\bar{S}$ is true) or a missed detection (i.e. $\bar{S}$ is chosen when S is true).

A well known result in hypothesis testing is that the minimum risk of error is given by the following decision rule :

$$\frac{p(z|S)}{p(z|\bar{S})} \underset{S}{\overset{\bar{S}}{\lessgtr}} \frac{1 - P_k(S)}{P_k(S)}. \quad (2)$$

In the recent paper from A. Hanjalic [6], the likelihood functions $p(z|S)$ and $p(z|\bar{S})$ and $P_k(S)$, the probability for the validity of S, have been specifically modelled for video shot boundaries detection. We will follow a similar modeling.

By plotting and analysing the shape of the distribution of the values of the dissimilarity profile within shots, the likelihood function $p(z|\bar{S})$ can be found to be correctly approximated in the family of exponential functions :

$$p(z|\bar{S}) = h_1 e^{-h_2 z}. \quad (3)$$

Using the same method with the distribution of the values at the shot boundaries, the likelihood function $p(z|S)$ can be found to belong to the family of Gaussian functions :

$$p(z|S) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\mu)^2}{2\sigma^2}}. \quad (4)$$

The parameters h_1, h_2, μ and σ are estimated using training data.

$P_k(S)$ is defined as the product of an "a priori" probability $P_k^a(S)$ that takes into account the length of the shot and the conditional probability $P_k(S|\phi(k))$ that depends of an additional information collected from the video.

$$P_k(S) = P_k^a(S)P_k(S|\phi(k)). \quad (5)$$

The "a priori" function will depend on the length of the shots. This will prevent our system to detect shot boundaries that are separated by an unrealistic number of frames. The "a priori" probability needs then to be equal to 0 immediately after the detection of a hardcut and reaches the not-informative value of 0.5 when a sufficient number of elapsed frames between the last shot detected and the frame k, $\lambda(k)$, is attained. It has been shown in [7] that the distribution of shot lengths across a large amount of motion pictures follows a Poisson function.

$$P_k^a(S) = \frac{1}{2}\sum_{w=0}^{\lambda(k)} \frac{\mu^w}{w!}e^{-\mu}. \quad (6)$$

In order to compute the conditional probability $P_k(S|\phi(k))$ for a boundary presence, we need a simple deterministic hardcut detector that will provide the function $\phi(k)$ that will vary between 0 and 100. We will use an adaptive thresholding method to find it. The conditional probability should be an increasing function that vary between 0 and 1. This should not be too sensitive when $\phi(k)$ has extreme values close to 0 or 100. Between these values, the transition should be smooth to avoid the rejection of good candidates and this is the reason why $P_k(S|\phi(k))$ can be chosen as :

$$P_k(S|\phi(k)) = \frac{1}{2}\left(1 + erf\left(\frac{\phi(k) - d}{\sigma_{erf}}\right)\right) \quad (7)$$

The parameters μ, d and σ_{erf} are estimated from the training data and erf is the error function computed as twice the integral of the Gaussian distribution with 0 mean and variance of $\frac{1}{2}$. The training data is a set of video as various as cinema, news report and television broadcast.

4. INFORMATION-BASED PARTITIONING OF ORDERED DATA

4.1. Properties and modeling of the color dissimilarity profile

The partitioning of our video is done by considering that a segment has been generated by a model. The choice of the model is constrained by the following criteria :

- we need a model able to fit the data during a homogeneous segment;
- we need an excellent detection performance in order to capture when dynamic of the video has significantly changed : the model should not fit discontinuities or any major changes in the temporal evolution;
- the model should be generic enough to be valid for any type of video.

We will use the cumulative sum of the dissimilarity profile for the information-based segmentation but only within continuous segments. If the evolution of the colors is homogeneous and the frame-by-frame dissimilarity is roughly constant, the cumulative sum of the dissimilarity profile is expected to have a linear behavior.

The model that we will use for a segment is then :

$$y_t^m(\theta) = a_1 t + a_0 + e_t \quad (8)$$

The additive error terms, e_t, are assumed to be $i.i.d$ and the error density $N(0, \sigma)$ for unknown σ. We use least square estimates of the linear coefficients.

An important property of this model for the user is the possibility to add Gaussian noise $N(0, \sigma_u)$ to e_t in order to control the partitioning. By adjusting one single parameter (the variance σ_u), the user can choose from a partitioning containing only one segment per shot (by setting a high value to σ_u) to the most accurate partitioning which is possible to get from the data (by setting $\sigma_u = 0$).

4.2. Partitioning

The segmentation problem is about finding the partitioning that best explains the data assuming a model y_t^m with different parameters $\theta = (a_0, a_1, \sigma)$ in each different segment.

Recently, a Minimum Message Length (MML) criterion has been used by L. J. Fitzgibbon [8] to infer the number of segments and the location of the cut-points from univariate temporal data using Fisher's DPA . The MML criterion has been experimentally proven to be more powerful to accurately locate the boundaries of the segments than other criteria such as the Minimum Description Length (MDL), the Bayesian Information Criteria (BIC) or Akaike's Information Criteria (AIC) by L. J. Fitzgibbon. The MML is based on the compact coding theory [9]. The idea is that the best explanation of the data is the one that provides the briefest encoding of a two-part message. The first part contains the information about the statistical model while the second part contains the remaining information needed about the data assuming the model. This is a quantification of the trade-off between the model complexity and the goodness of fit.

The partitioning $s = (s_0, ..., s_{G-1})$ containing G segments that maximizes the homogeneity of the evolution of the data according to the model y_t^m is also the one that minimizes the total message length. The minimum message length gives a natural way of choosing which partitioning is to be preferred using all the available data.

The message length formula used to calculate the expected length of a message which transmits the model and the data of the jth segment containing the data $y = (y_0, ..., y_n)$ can be approximated according to [10] by :

$$Mess(\theta)_j^k = -log(P_r) + \frac{m+2}{2} log(n-m-1) \\ -(m+1)log(\sigma) + \sum_{i=0}^{m} log(R_{a_i}) - log(f(y|\theta_j)) \quad (9)$$

where m is the degree of the polynomial ($m = 1$ in our case), R_{a_i} are the parameter's ranges of the two parameters a_0 and a_1 of equation (8). We specify $log(R_{a_0}) = log(R_{a_1}) = 10$ in our experiments. P_r is a prior information that we will design in order to meet our requirements. The "a priori" information will depend on the length of the segment to penalize the creation of too small partitions as in the equation (6).

$$P_r = \sum_{w=0}^{\lambda(k)} \frac{\mu^w}{w!} e^{-\mu}. \quad (10)$$

The parameter μ is chosen in order to reach a non-informative value after a given number of frames. The minus log-likelihood will be minimized when the model best fits the data. It is given by :

$$-log(f(y|\theta_j)) = nlog(\sqrt{2\pi}\sigma_j) + \frac{1}{\sigma_j^2} \sum_{t=1}^{n} (y_t - a_{1j}t - a_{0j})^2 \quad (11)$$

The total message length to minimize for the univariate sequence $(y)_k$ is given by :

$$Mess_{total}^k = log^*(G) + log(\binom{K-1}{G-1}) + \sum_{j=1}^{G} Mess(\theta)_j^k \quad (12)$$

4.3. Minimization

The problem is now to conduct the optimization in order to get the best partitioning of the ordered set of K numbers into G contiguous groups. This is a combinatorial problem and there are $\binom{K-1}{G-1}$ possibilities to explore.

An exhaustive way of searching for this partitioning has been solved in polynomial time by W. D. Fisher [11] using a Dynamic Programming Algorithm (DPA). The exhaustive search algorithm is based on the "Sub-optimisation Lemma" :

Lemma 4.1 *If $A_1 : A_2$ denotes a partition of a set A into two disjoint subsets A_1 and A_2, if P_1^* denotes a least square partition of A_1 into G_1 subsets and if P_2^* denotes a least square partition of A_2 into G_2 subsets, then, of the class of sub-partitions of $A_1 : A_2$ employing G_1 subsets over A_1 and G_2 subsets over A_2 a least square partition is $P_1^* : P_2^*$*

The time complexity of the DPA is reduced because the optimal solution is a combination of optimal solutions of subinstances. For a set of K numbers and a maximum number of groups G_{max}, the time complexity is $O(G_{max}.K^2)$.

The number G of partitions and the locations of the boundaries are then infered, and we know exhaustively that this partitioning and this number of partitions will maximize the homogeneity of the data in each partition according to our model.

The MML/DPA strategy presents two very interesting advantages over other segmentation techniques like agglomerative or greedy clustering strategies :

- It is a global and exhaustive approach : every possible partitioning is taken into account during the minimization process and there is then no risk to end in a local minima.
- No heuristic (no threshold) is needed to stop the clustering process. This is theoretically more satisfying.

5. APPLICATIONS

5.1. Keyframe selection

A keyframe is a simple, but also a very efficient way to represent a video sequence. Many authors [12], [13], [14] use clustering methods (sometimes with a time constraint) and select one keyframe per cluster. Another idea is to search efficiently for the least correlated frames of a video [15]. The problem with many of these methods is that the user needs to specify the number of keyframes as a stopping criterion rule.

We want to select the keyframes that are as much informative as possible. In our case, the clustering process has already taken place and the number of useful keyframes is equal to the number of homogeneous partitions that we have determined. In each homogeneous partition, the keyframe k will be chosen as the one that is the closest to all the other ones and therefore to the mean of the color histograms of the segment's frames $\bar{H}$ by minimizing this quantity :

$$\min_k D(H_k, \bar{H}) \quad (13)$$

where D is the Jeffrey divergence and H_k is the histogram of the k-th frame. The figures 1, 3 and 5 show results of keyframe selection for a set of classical examples. The figures 2 and 4 shows the temporal partitioning superposed to the video features.

5.2. Shot boundaries detection

The temporal segmentation in homogeneous segments has already brought us a lot of information about the structure

of the video, but there are too many segments in comparison of the number of shots that we should take into account in order to make the reverse-engineering of the video production process. It is important to make it correctly not to negatively influence higher-level analysis like video summarization for example.

The problem of shot boundaries detection is now reduced to the merging of the segments that are not due to a transition between shots. We will use a classical idea : if there is really a shot transition between the segments, the difference of the frames before and after the transition should be high. We want to compute this difference in a very discriminative way and we will take into account differences in color and spatial distribution of pixels information. If k is the frame number that separates the segments i and $i+1$, we define the distance $D_{seg}(i, i+1)$ between the frames $k-l$ and $k+l$ as a 2 components vector containing :

- the color block-histogram distance using the Jeffrey divergence in the RGB color space and 16 rectangular blocks;
- the mutual information of the frames as defined in [16];

where l is a parameter setting the minimum size of the transitions. We have averaged the vectors $D_{seg}(i, i+1)$ obtained for different value of l varying from 3 to 6 in our experiments to allow a reasonable maximum length for the transitions. The components of the vector $D_{seg}(i, i+1)$ are expected to be high if there is a shot boundary between the segments i and $i+1$ and low if there is no shot boundary. In order to find the best thresholds to separate each of these components, we use the K-means algorithm and specify that we are looking for 2 clusters only.

We will merge the segments when the vector $D_{seg}(i, i+1)$ belong to the class of the lowest values. We know after this post-processing step that only segments with high difference values before and after the transition are preserved.

We have experimented this method with 3 videos of the AIM corpus [1] using the evaluation framework of [17] and the results are given in the table 1. Each video contains around 200 transitions of every kind. There are many type of special-effects involved because the videos come from television.

The comparison of the tables 1 and 2 shows an advantage of the proposed framework. The recall and precision of our method are better than those of the CLIPS and LIMSI methods [18]. A better recall means there are a fewer number of missed detections while the better precision means that there are a fewer number of false detections. It shows that our method have better detection performance due to the information-based segmentation and that the merging process reduces appropriately the false detections.

[1]The AIM corpus has been developped within the French inter-laboratory research group ISIS and the French Institut National Audiovisuel (INA)

Detection	aim3	aim4	aim5
Accuracy	82.48	78.92	83
Recall	89.83	83.4	87
Precision	92.44	94.89	95.60

Tab. 1. Performances in percentage of the information-based shot boundaries detection algorithm in comparison with the ground truth (with 6 frames of tolerance for the accuracy of the location of the transitions)

It is also interesting to compare the tables 1 and 3 to see what brings the information-based segmentation in comparison with the simple detection of discontinuities described in the section 3. It shows that depending of the video concerned the detection of discontinuities can have very different performances whereas the information-based segmentation is more robust to the different type of videos and of transitions.

Detection	CLIPS	LIMSI
Accuracy	73.8	70
Recall	88.81	82.28
Precision	84.4	89.2

Tab. 2. Performances in percentage of the CLIPS and LIMSI [18] methods for a similar experiment and corpus when all type of transitions are taken into account

Detection	aim3	aim4	aim5
Accuracy	86.44	71.74	66
Recall	88.13	73.09	68.5
Precision	98.11	98.18	96.47

Tab. 3. Performances in percentage of the detection of discontinuities algorithm in comparison with the ground truth (with 6 frames of tolerance for the accuracy of the location of the transitions)

6. CONCLUSION

We have described an offline temporal segmentation algorithm using the global minimization of an information-based criterion. This approach is able to detect all types of transitions. We have also shown that the shot boundaries detection and the keyframe selection problems are highly sim-

plified and perform efficiently using our methodology. In the future, we will use these atomic temporal segments for video representation and characterization. A better characterization of the video segments is in our view the only way to improve the accuracy of the results shown in the table 1. We will also be interested in analysing trends and patterns through a video database for summarization and categorization of video collections.

7. AKNOWLEDGEMENTS

This work is supported by the EU project M4 - Multimodal Meeting Manager and the Swiss National Center of Competence IM2 - Interactive Multimedia Information Management.

8. REFERENCES

[1] Irena Koprinska and Sergio Carrato. Temporal video segmentation: A survey. *Signal Processing: Image Communication*, 16(5):477–500, 2001.

[2] U. Gargi, R. Kasturi, and S. Antani. Performance characterization and comparison of video indexing algorithms. In *Int. Conf. Computer Vision and Pattern Recognition (CVPR'98)*, pages 559–565, 1998.

[3] W. J. Heng and K. N. Ngan. Shot boundary refinement for long transition in digital video sequence. *IEEE transactions on multimedia*, 4(4), 2002.

[4] R. Lienhart. Comparison of automatic shot boundary detection algorithms. In *Storage and Retrieval for Still Image and Video Databases VII Proc. SPIE 3656-29*, 1999.

[5] U. Gargi, S. H. Strayer, U. Gargi, S. Antani, and R. Kasturi. An evaluation of color histogram based methods in video indexing. Technical report, Pennsylvania State University, 1996.

[6] A. Hanjalic. Shot-boundary detection: Unraveled and resolved. *IEEE transactions on circuits and systems for video technology*, 12(2), 2002.

[7] B. Salt. Statistical style analysis of motion pictures. *Film Quaterly*, 28:12–22, 1973.

[8] L. J. Fitzgibbon, L. Allison, and D. L. Dowe. Minimum message length grouping of ordered data. In H. Arimura and S. Jain, editors, *Proceedings of the Eleventh International Conference on Algorithmic Learning Theory (ALT2000)*, LNAI, pages 56–70, Berlin, 2000. Springer-Verlag.

[9] C. S. Wallace and P. R. Freeman. Estimation and inference by compact coding. *Journal of the Royal Statistical Society*, 49(3):240–265, 1987.

[10] R. A. Baxter and D. L. Dowe. Model selection in linear regression using the mml criterion. In J. A. Storer and M. Cohn, editors, *Proc. 4'th IEEE Data Compression Conference*, page 498, 1994.

[11] W. D. Fisher. On grouping for maximum homogeneity. *Journal of the American Statistical Association*, 53(284), 1958.

[12] M. S. Drew and J. Au. Video keyframe production by efficient clustering of compressed chromaticity signatures. In *ACM Multimedia '00*, pages 365–368, 2000.

[13] A. Girgensohn and J. S. Boreczky. Time-constrained keyframe selection technique. *Multimedia Tools and Applications*, 11(3):347–358, 2000.

[14] D. Gatica-Perez, A. Loui, and M.T. Sun. Finding structure in consumer videos by probabilistic hierarchical clustering. *IEEE Transactions on Circuits and Systems for Video Technologys*, 2003.

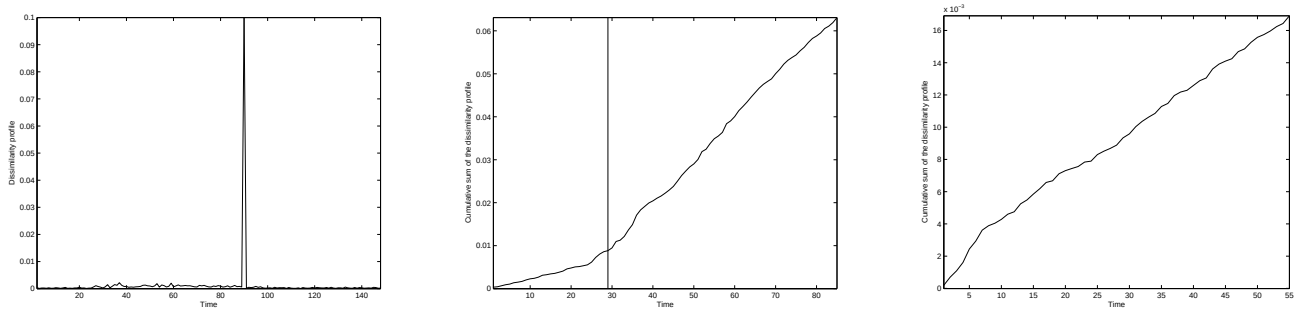
[15] N. D. Doulamis A. D. Doulamis. Stochastic search algorithms for content-based sampling of video sequences. In *14th IEEE International Conference on Digital Signal Processing*, 2002.

[16] Z. Cernekova, C. Nikou, and I. Pitas. Shot detection in video sequences using entropy-based metrics. In *IEEE 2002 International Conference on Image Processing 22th -25th Sept.*, 2000.

[17] R. Ruiloba, P. Joly, S. Marchand-Maillet, and Georges Quenot. Towards a standard protocol for the evaluation of video-to-shots segmentation algorithms. In *International Workshop in Content-Based Multimedia Indexing (CBMI)*, 1999.

[18] G. Quenot and P. Mulhem. Two systems for temporal video segmentation. In *European Workshop on Content Based Multimedia Indexing*, pages 187–194, 1999.

Fig. 1. Keyframes extracted from the 'tennis' sequence where the keyframes 1 and 2 are separated by a zoom-out and the keyframes 2 and 3 by a hardcut.



(a) (b) (c)

Fig. 2. (a) Dissimilarity profile of the 'tennis' sequence. (b) and (c) Cumulative sum of the dissimilarity profile and partitioning of the 'tennis' sequence within the shot 1 and the shot 2.

Fig. 3. Keyframes extracted from the 'ariel' sequence where the transitions are very smooth dissolve effects.

(a) (b)

Fig. 4. (a) Dissimilarity profile of the 'ariel' sequence. (b) Cumulative sum of the dissimilarity profile and partitioning of the 'ariel' sequence within the shot.

Fig. 5. Keyframes extracted from the ’soir3’ sequence where the very long sequence with the commentator is as expected represented by a single keyframe