Analysis and Re-Use of Videos in Educational Digital Libraries with Automatic Scene Detection

Lorenzo Baraldi(✉), Costantino Grana, and Rita Cucchiara

Dipartimento di Ingegneria "Enzo Ferrari", Università Degli Studi di Modena e Reggio Emilia, Via Vivarelli 10, 41125 Modena, MO, Italy
{lorenzo.baraldi,costantino.grana,rita.cucchiara}@unimore.it

Abstract. The advent of modern approaches to education, like Massive Open Online Courses (MOOC), made video the basic media for educating and transmitting knowledge. However, IT tools are still not adequate to allow video content re-use, tagging, annotation and personalization. In this paper we analyze the problem of identifying coherent sequences, called scenes, in order to provide the users with a more manageable editing unit. A simple spectral clustering technique is proposed and compared with state-of-the-art results. We also discuss correct ways to evaluate the performance of automatic scene detection algorithms.

Keywords: Scene detection · Performance evaluation · Spectral clustering

1 Introduction

In recent years, the research efforts in video access and video re-use have expanded their interest boundaries beyond traditional fields like news, web entertainment, and sport broadcasting to explore new areas, given the pervasive availability of huge amounts of digital footage. One of such emerging field is surely education that is a key-topic of many international research programs, like the European programs on Smart Communities and the 2020 European Digital Agenda, and that can benefit considerably in accessing the available digital material.

Indeed, many modern approaches to education try to engage the students with technological novelties, such as touch screens [3], hand and body pose recognition [2,10] or multimedia contents. In particular, Massive Open Online Courses (MOOC) already make use of video as the basic media for educating and transmitting knowledge. Moreover, recent educational projects rethink the concepts of the classical transmission model of the education, towards a socio-cultural-constructivist model where the massive use of video and multimedia content becomes the principal actor in the process of construction of new knowledge centered on the student in strict collaboration between broadcasting bodies,

D. Calvanese et al. (Eds.): IRCDL 2015, CCIS 612, pp. 155–164, 2016.
DOI: 10.1007/978-3-319-41938-1_16

content owners, teachers and the whole society [8]. For this aim, new instruments should be provided to each level of school for accessing and re-using media contents in different topics, allowing a personalized creation of knowledge, a sharing of multi-cultural practices and the assessment of new social experiences.

In the "Citt Educante" research project, in which we are involved, we are developing new solutions for the re-use of educational video production. The goal is to provide efficient tools for students to access the video content, creating their personalized educational experience on specific topics (e.g. geography or art) and across-topics, to share experiences by enriching the footage with user-generated content and data coming from web and social media. In this scenario, even if a huge amount of video from national broadcasting agencies is available and pedagogy researchers are trying to leverage this new possibility in education, the IT tools are still not adequate to allow video content re-use, tagging, annotation and personalization [4].

Nowadays, people can access video through web or specific apps, but it is difficult to find which section is really the one they want (e.g. a two minute scene withing a two hour program). Even if we know what is the part of interest, extracting and integrating it in our own presentation and re-using it in a suitable manner is still challenging. One basic necessary tool should allow an "access by scene" that improves the level of abstraction from single frame or shot to the scene, i.e. a conceptually meaningful and homogeneous element, composed by more than one shot. Unfortunately, most of the reusable content, owned by broadcast agencies, has not pre-defined sub-units and is not annotated. Therefore, we need accurate scene detection to identify coherent sequences (i.e. scenes) in videos, without asking manual segmentation to editors or publishers. The problem has been approached in the past in the literature with some promising, but not conclusive, results.

We present a novel proposal for scene segmentation, based on spectral clustering, which shows competitive results when compared to state-of-the-art methods. As well, also the broad concept of accuracy should be better defined for scene detection, especially when the goal is not only an algorithm comparison but a concrete result, which should be useful in many applications where a successive human interaction is expected, e.g. for browsing, tagging, selecting etc. In this case, for instance, the precise position of the cut is not important while skipping a scene and integrating it in another longer, preventing people (in our case students) to find a useful part of the video without seeing all the material, is more important. Thus we compare classical precision/recall measures with a better suited definition of coverage/overflow, which solves frequently observed cases in which the numeric interpretation would be quite different from the expected results by users.

The rest of this paper is organized as follows: Sect. 2 presents a summary of the existing approaches to scene detection and temporal clutering. In Sect. 3 we describe our algorithm; in Sect. 4 we discuss performance evaluation and in Sect. 5 experimentally evaluate them and show a sample use case.

2 Related Work

Video decomposition techniques aim to partition a video into sequences, like shots or scenes. Shots are elementary structural segments that are defined as sequences of images taken without interruption by a single camera. Scenes, on the contrary, are often defined as series of temporally contiguous shots characterized by overlapping links that connect shots with similar content [6]. Therefore, the fundamental goal of scene detection algorithms is to identify semantically coherent shots that are temporally close to each other. Most of the existing works can be roughly categorized into three categories: *rule-based methods*, that consider the way a scene is structured in professional movie production, *graph-based methods*, where shots are arranged in a graph representation, and *clustering-based methods*. They can rely on visual, audio, and textual features.

Rule-based approaches consider the way a scene is structured in professional movie production. Of course, the drawback of this kind of methods is that they tend to fail in videos where film-editing rules are not followed, or when two adjacent scenes are similar and follow the same rules. Liu *et al.* [7], for example, propose a visual based probabilistic framework that imitates the authoring process and detects scenes by incorporating contextual dynamics and learning a scene model. In [5], shots are represented by means of key-frames, thus, the first step of this method is to extract several key-frames from each shot: frames from a shot are clustered using the spectral clustering algorithm, color histograms as features, and the euclidean distance to compute the similarity matrix. The number of clusters is selected by applying a threshold Th on the eigenvalues of the Normalized Laplacian. The distance between a pair of shots is defined as the maximum similarity between key-frames belonging to the two shots, computed using histogram intersection. Shots are clustered using again spectral clustering and the aforesaid distance measure, and then labeled according to the clusters they belong to. Scene boundaries are then detected from the alignment score of the symbolic sequences.

In graph-based methods, instead, shots are arranged in a graph representation and then clustered by partitioning the graph. The Shot Transition Graph (STG), proposed in [13], is one of the most used models in this category: here each node represents a shot and the edges between the shots are weighted by shot similarity. In [9], color and motion features are used to represent shot similarity, and the STG is then split into subgraphs by applying the normalized cuts for graph partitioning. More recently, Sidiropoulos *et al.* [11] introduced a new STG approximation that exploits features automatically extracted from the visual and the auditory channel. This method extends the Shot Transition Graph using multimodal low-level and high-level features. To this aim, multiple STGs are constructed, one for each kind of feature, and then a probabilistic merging process is used to combine their results. The used features include visual features, such as HSV histograms, outputs of visual concept detectors trained using the Bag of Words approach, and audio features, like background conditions classification results, speaker histogram, and model vectors constructed from the responses of a number of audio event detectors.

We propose a simpler solution based on the spectral clustering approach, where we modify the standard spectral clustering algorithm in order to produce temporally consistent clusters.

3 A Spectral Clustering Approach

Our scene detection method generates scenes by grouping adjacent shots. Shots are described by means of color histograms, hence relying on visual features only: given a video, we compute a three-dimensional histogram of each frame, by quantizing each RGB channel in eight bins, for a total of 512 bins. Then, we sum histograms from frames belonging to the same shot, thus obtaining a single L_1-normalized histogram for each shot.

In contrast to other approaches that used spectral clustering for scene detection, we build a similarity matrix that jointly describes appearance similarity and temporal proximity. Its generic element κ_{ij}, defines the similarity between shots $\mathbf{x}_i$ and $\mathbf{x}_j$ as

$$\kappa_{ij} = \exp\left(-\frac{d_1^2(\psi(\mathbf{x}_i), \psi(\mathbf{x}_j)) + \alpha \cdot d_2^2(\mathbf{x}_i, \mathbf{x}_j)}{2\sigma^2}\right) \tag{1}$$

where $\psi(\mathbf{x}_i)$ is the normalized histogram of shot $\mathbf{x}_i$, d_1^2 is the Bhattacharyya distance and $d_2^2(\mathbf{x}_i, \mathbf{x}_j)$ is the normalized temporal distance between shot $\mathbf{x}_i$ and shot $\mathbf{x}_j$, while the parameter α tunes the relative importance of color similarity and temporal distance. To describe temporal distance between frames, $d_2^2(\mathbf{x}_i, \mathbf{x}_j)$ is defined as

$$d_2^2(\mathbf{x}_i, \mathbf{x}_j) = \frac{|m_i - m_j|}{l} \tag{2}$$

where m_i is the index of the central frame of shot $\mathbf{x}_i$, and l is the total number of frames in the video. The spectral clustering algorithm is then applied to the similarity matrix, using the Normalized Laplacian and the maximum eigen-gap criterion to select k:

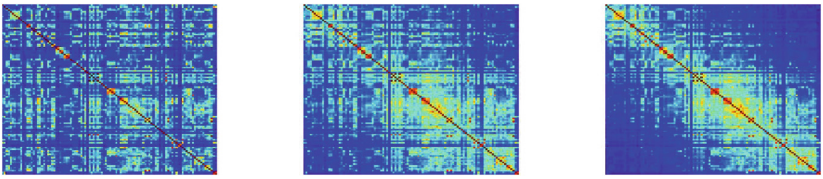
$$k = \arg\max\left(\lambda_{i+1} - \lambda_i\right) + 1 \tag{3}$$

where λ_i is the i-th eigenvalue of the Normalized Laplacian.

As shown in Fig. 1, the effect of applying increasing values of α to the similarity matrix is to raise the similarities of adjacent shots, therefore boosting the temporal consistency of the resulting groups. Of course, this does not guarantee a completely temporal consistent clustering (i.e. some clusters may still contain non-adjacent shots); at the same time, too high values of α would lead to a segmentation that ignores color dissimilarity. The final scene boundaries are created between adjacent shots that do not belong to the same cluster.

4 Evaluating Scene Segmentation

The first possibility to evaluate the results of a scene detection algorithm is to count correctly and wrongly detected boundaries, without considering the



(a) $\alpha = 0$ (b) $\alpha = 0.5$ (c) $\alpha = 1$

Fig. 1. Effect of α on similarity matrix κ_{ij}. Higher values of α enforce connections between near shots and increase the quality of the detected scenes (best viewed in color). (Color figure online)

Fig. 2. Two consecutive scenes from the RAI dataset.

temporal distance between a ground truth cut and the nearest detected cut. The most used measures in this context are precision and recall, together with the F-Score measure, that summarizes both. Precision is the ratio of the number

of correctly identified scenes boundaries to the total number of scenes detected by the algorithm. Recall is the ratio of the number of correctly identified boundaries to the total number of scenes in the ground truth.

Of course this kind of evaluation does not discern the seriousness of an error: if a boundary is detected one shot before or after its ground truth position, an error is counted in recall as if the boundary was not detected at all, and in precision as if the boundary was put far away. This issue appears to be felt also by other authors, with the result that sometimes a tolerance factor is used. For example, [9] uses a *best match* method with a sliding window of 30 s, so that a detected boundary is considered correct if it matches a ground truth boundary in the sliding window.

To deal with these problems, Vendrig *et al.* [12] proposed the Coverage and Overflow measures. Coverage $\mathcal{C}$ measures the quantity of shots belonging to the same scene correctly grouped together, while Overflow $\mathcal{O}$ evaluates to what extent shots not belonging to the same scene are erroneously grouped together. Formally, given the set of automatically detected scenes $\mathbf{s} = [\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_m]$, and the ground truth $\tilde{\mathbf{s}} = [\tilde{\mathbf{s}}_1, \tilde{\mathbf{s}}_2, ..., \tilde{\mathbf{s}}_n]$, where each element of $\mathbf{s}$ and $\tilde{\mathbf{s}}$ is a set of shot indexes, the coverage $\mathcal{C}_t$ of scene $\tilde{\mathbf{s}}_t$ is proportional to the longest overlap between $\mathbf{s}_i$ and $\tilde{\mathbf{s}}_t$:

$$\mathcal{C}_t = \frac{\max_{i=1...,m} \#(\mathbf{s}_i \cap \tilde{\mathbf{s}}_t)}{\#(\tilde{\mathbf{s}}_t)} \tag{4}$$

where $\#(\mathbf{s}_i)$ is the number of shots in scene $\mathbf{s}_i$. The overflow of a scene $\tilde{\mathbf{s}}_t$, $\mathcal{O}_t$, is the amount of overlap of every $\mathbf{s}_i$ corresponding to $\tilde{\mathbf{s}}_t$ with the two surrounding scenes $\tilde{\mathbf{s}}_{t-1}$ and $\tilde{\mathbf{s}}_{t+1}$:

$$\mathcal{O}_t = \frac{\sum_{i=1}^{m} \#(\mathbf{s}_i \setminus \tilde{\mathbf{s}}_t) \cdot \min(1, \#(\mathbf{s}_i \cap \tilde{\mathbf{s}}_t))}{\#(\tilde{\mathbf{s}}_{t-1}) + \#(\tilde{\mathbf{s}}_{t+1})} \tag{5}$$

The computed per-scene measures can then be aggregated into values for an entire video as follows:

$$\mathcal{C} = \sum_{t=1}^{n} \mathcal{C}_t \cdot \frac{\#(\tilde{\mathbf{s}}_t)}{\sum \#(\tilde{\mathbf{s}}_i)}, \quad \mathcal{O} = \sum_{t=1}^{n} \mathcal{O}_t \cdot \frac{\#(\tilde{\mathbf{s}}_t)}{\sum \#(\tilde{\mathbf{s}}_i)}. \tag{6}$$

Finally, an F-Score measure can be defined to combine Coverage and Overflow in a single measure, by taking the harmonic mean of $\mathcal{C}$ and $1 - \mathcal{O}$.

5 Evaluation

We evaluate the aforesaid measures and algorithms on a collection of ten challenging broadcasting videos from the Rai Scuola video archive[1], mainly documentaries and talk shows. Shots have been obtained running the state of the art shot detector of [1] and manually grouped into scenes by a set of

[1] http://www.scuola.rai.it.

human experts to define the ground truth. Our dataset and the corresponding annotations are available for download at http://imagelab.ing.unimore.it/files/RaiSceneDetection.zip.

We reimplemented the approach in [5] and used the executable of [11] provided by the authors[2]. The threshold Th of [5] was selected to maximize the performance on our dataset, and α was set to 0.05 in all our experiments.

Figure 3 shows the results of the compared methods on a frame sequence from our dataset.

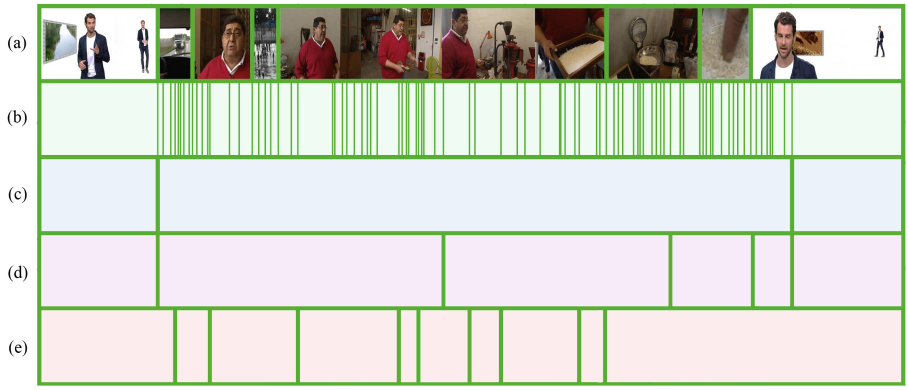


Fig. 3. Samples results on our dataset. Row (a) shows the ground-truth segmentation, (b) the individual shots boundaries, row (c) shows the results of our method, (d) those of [11] and (e) those of [5] (best viewed in color). (Color figure online)

Table 1. Performance comparison on the RAI dataset using the boundary level measures (Precision, Recall, F-Score)

Video	Spectral Clustering			Chasanis *et al.* [5]			Sidiropoulos *et al.* [11]		
	F-Score	Precision	Recall	F-Score	Precision	Recall	F-Score	Precision	Recall
V_1	0.12	0.09	0.17	0.25	0.20	0.33	**0.29**	0.25	0.33
V_2	**0.36**	0.27	0.55	0.00	0.00	0.00	0.30	0.33	0.27
V_3	**0.37**	0.29	0.53	0.13	0.13	0.13	0.31	0.36	0.27
V_4	**0.30**	0.23	0.43	0.10	0.10	0.10	0.22	0.50	0.14
V_5	**0.44**	0.31	0.75	0.00	0.00	0.00	0.36	0.31	0.42
V_6	0.18	0.10	0.75	0.00	0.00	0.00	**0.36**	0.29	0.50
V_7	**0.18**	0.33	0.13	0.00	0.00	0.00	0.13	0.13	0.13
V_8	0.10	0.06	0.27	0.13	0.10	0.18	**0.21**	0.25	0.18
V_9	**0.25**	0.16	0.62	0.00	0.00	0.00	0.21	0.33	0.15
V_{10}	0.23	0.15	0.60	**0.26**	0.38	0.20	0.19	0.33	0.13
Average	0.25	0.20	0.48	0.09	0.09	0.09	**0.26**	0.31	0.25

[2] http://mklab.iti.gr/project/video-shot-segm.

Table 2. Performance comparison on the RAI dataset using the Shot level measures (Coverage, Overflow and F-Score)

Video	Spectral Clustering			Chasanis *et al.* [5]			Sidiropoulos *et al.* [11]		
	F-Score	$\mathcal{C}$	$\mathcal{O}$	F-Score	$\mathcal{C}$	$\mathcal{O}$	F-Score	$\mathcal{C}$	$\mathcal{O}$
V_1	0.64	0.81	0.48	0.70	0.64	0.24	**0.72**	0.84	0.37
V_2	**0.68**	0.61	0.22	0.36	0.80	0.77	0.59	0.85	0.55
V_3	**0.65**	0.68	0.38	0.58	0.73	0.52	0.58	0.90	0.57
V_4	**0.74**	0.69	0.22	0.50	0.65	0.60	0.33	0.94	0.80
V_5	**0.77**	0.68	0.11	0.25	0.93	0.86	0.66	0.76	0.41
V_6	0.51	0.37	0.17	0.18	0.89	0.90	**0.71**	0.77	0.34
V_7	0.30	0.97	0.82	0.37	0.70	0.75	**0.51**	0.78	0.62
V_8	0.59	0.53	0.33	**0.62**	0.57	0.32	0.45	0.88	0.70
V_9	**0.67**	0.55	0.15	0.27	0.87	0.84	0.43	0.92	0.72
V_{10}	**0.57**	0.42	0.12	0.54	0.91	0.62	0.44	0.94	0.71
Average	**0.61**	0.63	0.30	0.44	0.77	0.64	0.54	0.86	0.58

Tables 1 and 2 compare the two different approaches using Boundary level and Shot level performance measures. As show in Table 1, detected boundaries rarely correspond to ground truth boundaries exactly, therefore leading to poor results in terms of precision and recall, even when considering a recent and state-of-the-art approach like [11].

As expected, the two measures behave differently and there is not a complete agreement among them: [5] performs worse than the other two methods according to both measures, while [11] performs equal to the spectral clustering approach with boundary level measures, but slightly worse than the spectral clustering approach according to shot level measures.

Fig. 4. Effective video browsing using our algorithm. Users can visualize a summary of the content by means of the extracted scenes.

Detected scenes, finally, can be used as an input for video browsing or re-using software. As an example, we built a web-based browsing interface for broadcasting videos (see Fig. 4) where users can visualize a summary of the content by means of the extracted scenes. Scenes are represented with key-frames in a timeline fashion, and when a particular scene is selected, all its shots are unfolded. To ease the browsing even more, most frequent words, obtained from the transcript of the audio, are reported under each scene. Users can jump from one part of the video to another by clicking on the corresponding scene or shot.

6 Conclusions

We investigated the problem of evaluating scene detection algorithms with tests conducted on two different performance measures and on three different and recent approaches to scene segmentation. Results show that the problem of scene detection is still far from being solved, and that simple approaches like the suggested spectral clustering technique can sometimes achieve equivalent or better results than more complex methods.

Acknowledgments. This work was carried out within the project "Città educante" (ctn01_00034_393801) of the National Technological Cluster on Smart Communities cofunded by the Italian Ministry of Education, University and Research - MIUR.

References

1. Apostolidis, E., Mezaris, V.: Fast shot segmentation combining global and local visual descriptors. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6583–6587 (2014)
2. Baraldi, L., Paci, F., Serra, G., Benini, L., Cucchiara, R.: Gesture recognition in ego-centric videos using dense trajectories and hand segmentation. In: Proceedings of 10th IEEE Embedded Vision Workshop (EVW). Columbus, Ohio, June 2014
3. Battenberg, J.K., Merbler, J.B.: Touch screen versus keyboard: a comparison of task performance of young children. J. Spec. Educ. Technol. **10**(2), 24–28 (1989)
4. Bertini, M., Del Bimbo, A., Serra, G., Torniai, C., Cucchiara, R., Grana, C., Vezzani, R.: Dynamic pictorially enriched ontologies for video digital libraries. IEEE MultiMedia Mag. **16**(2), 41–51 (2009)
5. Chasanis, V.T., Likas, C., Galatsanos, N.P.: Scene detection in videos using shot clustering and sequence alignment. IEEE Trans. Multimedia **11**(1), 89–100 (2009)
6. Hanjalic, A., Lagendijk, R.L., Biemond, J.: Automated high-level movie segmentation for advanced video-retrieval systems. IEEE Trans. Circ. Syst. Vid. Technol. **9**(4), 580–588 (1999)
7. Liu, C., Wang, D., Zhu, J., Zhang, B.: Learning a contextual multi-thread model for movie/tv scene segmentation. IEEE Trans. Multimedia **15**(4), 884–897 (2013)
8. Mascolo, M.F.: Beyond student-centered and teacher-centered pedagogy: teaching and learning as guided participation. Pedagogy Hum. Sci. **1**(1), 3–27 (2009)
9. Rasheed, Z., Shah, M.: Detection and representation of scenes in videos. IEEE Trans. Multimedia **7**(6), 1097–1105 (2005)

10. Serra, G., Camurri, M., Baraldi, L., Benedetti, M., Cucchiara, R.: Hand segmentation for gesture recognition in ego-vision. In: Proceedings of ACM Multimedia International Workshop on Interactive Multimedia on Mobile and Portable Devices (IMMPD), Barcelona, Spain, October 2013
11. Sidiropoulos, P., Mezaris, V., Kompatsiaris, I., Meinedo, H., Bugalho, M., Trancoso, I.: Temporal video segmentation to scenes using high-level audiovisual features. IEEE Trans. Circ. Syst. Vid. Technol. **21**(8), 1163–1177 (2011)
12. Vendrig, J., Worring, M.: Systematic evaluation of logical story unit segmentation. IEEE Trans. Multimedia **4**(4), 492–499 (2002)
13. Yeung, M.M., Yeo, B.L., Wolf, W.H., Liu, B.: Video browsing using clustering and scene transitions on compressed sequences. In: IS&T/SPIE's Symposium on Electronic Imaging: Science & Technology, pp. 399–413 (1995)