DETECTION OF TV COMMERCIALS

Alberto Albiol , María José Ch. Fullà, Antonio Albiol

Technical University of Valencia, Spain
{alalbiol,machaful,aalbiol}@dcom.upv.es

Luis Torres

Technical University of Catalonia, Spain
luis@gps.tsc.upc.es

ABSTRACT

This paper presents a system that labels TV shots either as commercial or program shots. The system uses two observations: logo presence and shot duration. This observations are modeled using HMM and the Viterbi decoder is finally used for shot labeling. The system has been tested on several hours of real video achieving more than 99% of correct labeling.

1. INTRODUCTION

Automatic detection of TV advertisements is a topic with many practical applications. For instance, from the point of view of a TV end-user, it could be useful to avoid commercials in personal recordings. In another possible scenario, a TV-viewer could make *zapping* during commercials receiving a notification from the system when the commercial break has finished. But commercials detection has also many practical applications from the point of view of publicity companies and public institutions. For instance, it could be used to make commercial summaries which could be used by publicity companies to verify if the the TV station is broadcasting advertisements at the right schedules. Public institutions could also benefit from commercials detection to monitor that TV stations follows the corresponding legal regulations and they are not broadcasting an excessive number of commercials.

Most previous works on commercial break detection [1, 2] have based their strategies in studying the relation between audio silences and black frames as an indicator of commercials boundaries. The analysis is performed in either compressed [1] or uncompressed [2] domains. In [3] also specific country regulations about commercials broadcast is used as a further clue. Another interesting approach is presented in [4], where overlaid text trajectories are used as a clue to detect commercial breaks. The idea here is that overlaid text (if any) usually remains more stable during the program time than in the case of commercials.

This work has been partially supported by the grants TIC2001-0442, TIC2002-02469 of the Spanish Government and the US-Spain Joint Commission for Scientific and Technological Cooperation

Our approach to commercial detection relies on two simple observations to label each video shot as a *Commercial* or *Program* shot. The first observation is based on the fact that TV logos are removed during commercials (at least in the Spanish broadcasts). The second observation stems from the fact that video shots tend to have a shorter duration within commercials. These observations are modeled using HMM and the Viterbi algorithm is finally used to label each shot. One advantage of the proposed scheme is that it can be easily extended with more different observations, such as differences on the audio volume. However, in this paper, we only use the two previous observations.

The system proposed in this paper, can be regarded as a three-state machine. The names of these states are: *Inicialization*, *Commercial* and *Program*. Changes between these states only occur at the shot bounds, and obviously, shots are labeled according to the current state. Fig. 1 sketches the general flowchart of the state machine.

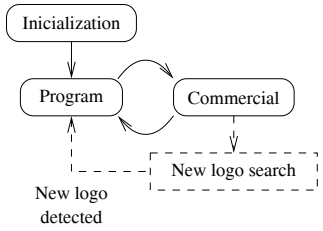

Fig. 1. General block diagram of the commercial detection system

The *Initialization* state occurs during the system set up only. The goal of this state is to extract a binary mask that indicates the region where the TV logo is placed. Details about the logo search algorithm are presented in Section 2. Whenever a logo mask is detected the system changes to the *Program* state. From this point on, the system is always either in the *Program* or *Commercial* state. Transitions between these two states are based on a Viterbi decoder that takes shot-based measurements as observations as explained in Section 3. An important requirement of the system is that it has to be robust to changes in the logo position (sometimes the logo moves to a different corner) or changes in the

logo pattern. To accomplish this requirement, the system starts a parallel process whenever the current state changes to *Commercial*. The goal of this process is to check if a new logo mask is being used. In case that a new mask is found the system is forced to change back to the *Program* state. If no new logo mask is found and the Viterbi decoder changes the state back to *Program* this parallel process is stopped.

The paper finishes with some results and conclusions presented in Section 4.

2. LOGO MASK EXTRACTION

This section describes our algorithm used to detect TV logos by the *Inicialization* and *Commercial* states. In general, TV logos can be grouped into opaque, transparent or animated. In this paper, we have only considered the case of opaque and transparent logos. This is justified because currently animated logos are not as frequent as the other types. However, we will see that the proposed approach is general enough to be used even with animated logos.

Initially, it may seem obvious that opaque logos are the easiest to detect since their pixel values remain constant (except for the noise). In this case, a variance analysis of the pixel values can be used to detect and track a logo mask. Once the logo mask is detected, tracking can be performed using a similarity measurement between the logo pattern and the pixels under the mask. However, this approach has two main drawbacks. First, it is too sensitive to small changes in the logo position (1-2 pixels produced by synchronization errors). Second, many logos that might look like opaque are actually transparent. This fact yields higher logo pixel variances that reduce the detection and tracking performance.

For these reasons we decided to come up with a new approach to logo detection general enough to be used with any type of logos. In this new approach we intuitively say that a logo exists if we can find an area in the image with *stable* contours. Notice that in this definition we focus on the area containing the contours and not in the contours themselves, therefore this definition also applies to the case of animated logos, considering the area that contains the moving contours. One advantage of our logo detection approach compared to other more sophisticated (and probably more reliable) pattern recognition techniques is that it does not requiere any supervised training and can easily used for any type of logos without human interaction.

The logo search is restricted to the four corners as shown in Fig. 2. For each corner a separate search is conducted and the process is stopped when at least one logo has been found. The process for one of the corners is summarized in Fig. 3. First, shot detection is used to extract one frame per shot. Let F_i be that frame where i indicates the number of shots processed. Then, the gradient of F_i is taken and the

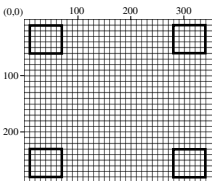

Fig. 2. The boxes indicate the areas where logo search is performed using CIF resolution

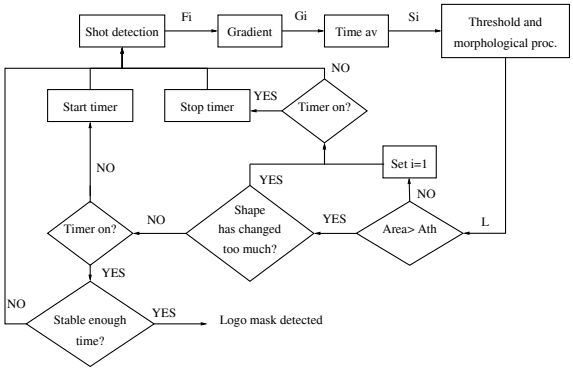

Fig. 3. Blocks diagram used to detect TV logos

result is time averaged:

$$S_i = \frac{i-1}{i} S_{i-1} + \frac{G_i}{i} \tag{1}$$

where G_i is the gradient image of F_i. The next step includes thresholding of S_i and morphological processing to reduce spurious pixels and fill holes. Morphological processing includes operations such as closing, opening and morphological area filtering [5]. The result of the morphological processing is a binary mask $\mathcal{L}_i$ as shown in Fig. 4.

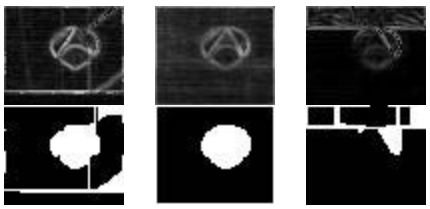

Fig. 4. Examples of logo mask extraction. The first row shows the time averaged gradient $\mathcal{S}_n$. The second row shows the corresponding masks. (see text for explanation).

Once the binary mask $\mathcal{L}_i$ has been obtained, two different tests are performed to check if the detection process has concluded. The first test checks if the area of the binary mask is over some threshold. If the area of the binary mask is too small as in Fig. 4.c, the search procedure is reset (i=1). This is a common situation when the search procedure has

started during the commercials. The second test checks that the mask shape remains stable for a specified time (at least 3 min.). This test guarantees that enough data has been gathered to extract the logo mask, and logos present during commercial times will be discarded. Fig. 4.a shows an example where too few frames have been processed. As the number of frames processed increases the logo shape stabilizes and finally a logo mask can be extracted as in Fig. 4.b.

3. SHOT LABELING

As introduced in Section 1, shot labeling is based on two observations shot duration (T_n) and logo presence (L_n), where n indicates the shot number.

Logo presence is measured for each shot using only one frame (the last one). Let $\mathcal{G}_n(i,j)$ be the result of the gradient of this frame and let $\mathcal{L}(i,j)$ be the logo mask obtained as described in the previous section, where $\mathcal{L}(i,j) = 1$ for the logo pixels and zero elsewhere. Then, L_n is computed as the mean value of the image gradient under the logo mask given by the following equation:

$$L_n = \frac{\sum_{i,j} \mathcal{G}_n(i,j).\mathcal{L}(i,j)}{\sum_{i,j} \mathcal{L}(i,j)} \qquad (2)$$

where the summatories are extended to all image pixels. A possible question at this point might be why using only one frame per shot. The answer is twofold. First, to reduce the computational burden. Second, preliminary examples have shown that using shot-based measurements did not produce a significant increase in the system performance, since as we shall see in Section 4 errors have different sources. An example of a shot-based measurement might be the average of the image gradients under the logo mask for every frame within the shot.

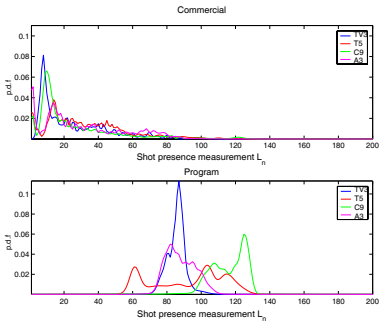

Fig. 5. p.d.f of the logo presence measurement L_n for commercials (above) and program time (below) extracted from four different TV stations.

Figures 5 and 6 show the histograms of the logo presence measurement L_n and the shot duration T_n. This histograms have been extracted from four different TV stations. From the figures, it is easy to conclude that L_n is more useful to the labeling process since the distributions of $H(L_n|p)$ and $H(L_n|c)$ are farther apart (p and c indicate the state *Program* and *Commercial* respectively). Actually, our experiments have shown that the shot duration observation is only useful for labeling programs like talk shows where usually long shots are performed.

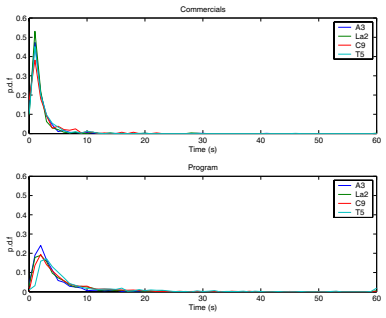

Fig. 6. Shot duration p.d.f for commercials (above) and program time (below) extracted from four TV stations.

As introduced in Section 1, the process is modeled using a HMM with two states, and then, the Viterbi algorithm is used to perform the shot labeling. Following the notation as in [6], a HMM is characterized by a set of parameters:

$$\lambda = (A, B, \pi) \qquad (3)$$

where A is the transition matrix, $\pi = \{\pi_c, \pi_p\}$ are the a priori probabilities for each state, and $B = \{b_c(O_n), b_p(O_n)\}$ are the observation probability distributions for each state given the observation $O_n = \{S_n, L_n\}$. The set of parameters λ has been estimated from more than 10 hours of manually labeled TV recordings. Using this data, the values of π are estimated as the percentage of time corresponding either to program or commercial, the matrix transition A is obtained using the mean number of shots per program or commercial block, and the density functions B are modeled using multivariate Gaussian Mixture Distributions (GMM) using diagonal covariance matrices. The parameters for each GMM are obtained using the EM algorithm. Fig. 7 shows an example with the probabilities $b_c(O_n)$, $b_p(O_n)$ input to the Viterbi decoder, where it can be seen how the probabilities effectively change depending on the underlying state.

4. RESULTS

The system described in this paper has been tested on 6 different Spanish TV stations. Some of them use opaque logos and other transparent. To evaluate the results we have defined three parameters:

- False acceptance (FA), percentage of the time that the system miss-classifies programs.

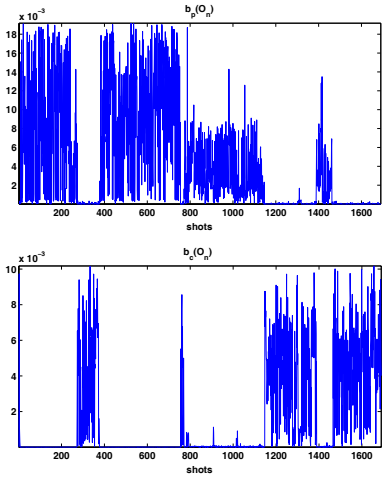

Fig. 7. Conditional probability values $b_c(O_n), b_p(O_n)$ input to the Viterbi decoder.

- False reject (FR), percentage of the time that the system miss-classifies commercials.
- Correct classification, percentage of the time that the system makes the correct classification.

TV Station	Total time	% FA	% FR	% Correct
Tele5	7161 s.	0.43	0.05	99.52
Ant3	7149 s.	0.21	0.26	99.53
C9	7091 s.	0.01	0.39	99.60
Tve1	7158 s.	0.15	0.25	99.60
La2	7018 s.	0.37	0.05	99.58
Tv3	7173 s.	0.6	0.06	99.34

Table 1. Labeling results obtained for 6 TV stations

The results obtained using the previous definitions are gathered in Table 1. Fig. 8 shows the outputs of the Viterbi decoder for two different examples. In Fig. 8.a the output obtained for the last sequence of Table 1 is shown. It can be seen that in this example the system makes the correct labeling for most of the time. The two errors found in this example are common to the other sequences. The missed commercial break (false reject) was caused by a brand logo placed in the same location that the program logo (see Fig 9.a). Next a false acceptance error is found. In this case the logo was very similar to the image background yielding too small values of L_n (see Fig 9.b). Fig. 8.b shows the robustness of the system to changes in the logo position. In this example, the logo changes its position from one corner to the other around the second 4000. Initially, the system changes its state to *Commercial* and starts the logo search process as described in Section 1. It can be seen that after a short time the system finds the new logo and then changes back to the *Program* state making the right decision.

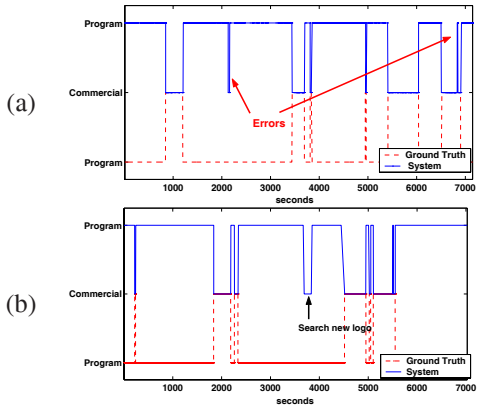

Fig. 8. Output of the Viterbi decoder: a) for the TV3 channel of Table 1, b) when the TV station is changed.

(a) (b)

Fig. 9. Examples of errors: a) false reject, b) false acceptance.

5. REFERENCES

[1] D. A. Sadlier et al., "Automatic TV advertisement detection from mpeg bitstream," *Journal of the Patt. Rec. Society*, vol. 35, no. 12, pp. 2–15, Dec. 2002.

[2] Y. Li and C.-C. Jay Kuo, "Detecting commercial breaks in real TV program based on audiovisual information," in *SPIE Proc. on IMMS, vol 421*, Nov. 2000.

[3] R. Lienhart, C. Kuhmnch, and W. Effelsberg, "On the detection and recognition of television commercials," in *Proc. IEEE Conf. on MCS*, Otawa, Canada, 1996.

[4] N. Dimitrova, "Multimedia content analysis: the next wave," in *Proc. of the 2nd CIVR*, Illinois, USA, Aug. 2003.

[5] L. Vincent, "Morphological are openings and closings for grayscale images," in *Shape in picture, NATO Workshop*, Driebergen, Setp. 1992.

[6] L. Rabiner, "A tutorial on hidden markow models and selected applications in speech recognition," *Proc. of the IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.