applied sciences

MDPI

Article

Automatic TV Logo Identification for Advertisement Detection without Prior Data

Pedro Carvalho [1,2,*], Américo Pereira [1,3] and Paula Viana [1,2]

[1] Centre for Telecommunications and Multimedia at INESC TEC—Institute for Systems and Computer Engineering, Technology and Science, 4200-465 Porto, Portugal; americo.j.pereira@inesctec.pt (A.P.); paula.viana@inesctec.pt (P.V.)
[2] School of Engineering, Polytechnic of Porto, 4249-015 Porto, Portugal
[3] Faculty of Engineering, University of Porto, 4200-465 Porto, Portugal
[*] Correspondence: pedro.m.carvalho@inesctec.pt; Tel.: +351-222094000

Abstract: Advertisements are often inserted in multimedia content, and this is particularly relevant in TV broadcasting as they have a key financial role. In this context, the flexible and efficient processing of TV content to identify advertisement segments is highly desirable as it can benefit different actors, including the broadcaster, the contracting company, and the end user. In this context, detecting the presence of the channel logo has been seen in the state-of-the-art as a good indicator. However, the difficulty of this challenging process increases as less prior data is available to help reduce uncertainty. As a result, the literature proposals that achieve the best results typically rely on prior knowledge or pre-existent databases. This paper proposes a flexible method for processing TV broadcasting content aiming at detecting channel logos, and consequently advertising segments, without using prior data about the channel or content. The final goal is to enable stream segmentation identifying advertisement slices. The proposed method was assessed over available state-of-the-art datasets as well as additional and more challenging stream captures. Results show that the proposed method surpasses the state-of-the-art.

Keywords: computer vision; logo detection; advertisement identification

check for updates

Citation: Carvalho, P.; Pereira, A.; Viana, P. Automatic TV Logo Identification for Advertisement Detection without Prior Data. *Appl. Sci.* **2021**, *11*, 7494. https://doi.org/10.3390/app11167494

Academic Editor: Byung-Gyu Kim

Received: 7 July 2021
Accepted: 9 August 2021
Published: 15 August 2021

1. Introduction

Audiovisual segments containing advertisements have long been inserted in TV broadcasts and have become pervasive to almost any multimedia content distribution service. In the particular case of television broadcasting, advertisements represent a significant source of revenue, with the ads being intermingled with a channel's linear programming. These ads, while short, may have different duration and visual characteristics, making it difficult to identify common traits. Moreover, there may also be self-advertisements where the channels announce content to come.

Several actors in the content distribution and consumption value chain would benefit from being able to automatically identify advertisement segments. Broadcasters and other content providers could benefit from making use of automatic detection modules to identify segments with advertisements, enabling their removal or substitution with others before content repurposing to augment revenues. Companies that bought time for transmitting their ads can use the detection modules to verify that the contracted conditions have been fulfilled by the broadcaster without the need of a cumbersome and costly manual inspection. Even consumers can benefit by having the detection modules running over stored content to remove the advertisements for a more enjoyable experience.

The audiovisual richness of advertisements is very large and ever-changing to cope with society and costumers demands, and advertising companies are pressed for new creative ideas. This makes it difficult to identify common traits in advertisements. Moreover, in the particular case of TV broadcasting, both channel characteristics and the way advertisements are inserted may vary from channel to channel, and these may also change depending on national context and type of channel (e.g., generic, news, and movies).

Over the years, there have been several government initiatives to regulate how advertisements should be inserted in TV linear programming. As a result, some proposals in the literature take advantage of such rules. However, these are often limited to a national scope, change over time, or cease to be adhered to. In 2010, with the aim of standardizing some general features advertisement segments in TV broadcasting should cohere to, the European Union published a set of recommendations that should be followed by the broadcasters of the member states [1]. In this directive, Article 19 indicates that television broadcasters should impose a set of rules on the broadcasted advertisements, so that (1) they are clearly recognizable and distinguishable from editorial content and (2) they should be distinct from other contents by optical and/or acoustic and/or spatial means. While providing base guidelines, the recommendations give room for each member state to define their own set of rules. In line with such recommendations, some broadcasters seek to differentiate linear content from advertisements by introducing a station logo on the transmitted content. This behavior resulted in the proposal of advertisement detection algorithms based on the identification of the channel logo. However, these proposals typically make use of prior knowledge by targeting specific channels or requiring databases of known logos to effectively work. In this paper, we propose an advertisement detector for television content based on logo identification without requiring any prior data. Moreover, the main target scenario also includes the introduction of as little delay as possible in the classification of advertisement segments, encompassing operation over live content. The algorithm automatically identifies suitable regions for the presence of the logo, followed by an automatic online training of a classifier. The contributions of the paper are twofold: First, state-of-the-art proposals were reviewed, as well as corresponding datasets (when available), and additional streams were collected and manually annotated to perform an analysis of different features present. Second, a new algorithm was developed to detect logos in television broadcast content, even in the presence of partial occlusion and without making use of prior data. The algorithm was assessed over the datasets demonstrating superior capacity with regards to the state-of-the-art.

The remainder of this article is structured as follows. Section 2 describes and compares advertisement detection methods from the literature, providing an analysis of the problems that are generally found. In Section 3, we detail the created television dataset, providing an analysis of the different features that can be found on advertisements and programs as well as an accounting of the most relevant aspects that can be used to differentiate advertisements from program content. Section 4 presents the proposed and implemented methodology for an automatic advertisement detection. The evaluation methodology and the obtained results are discussed on Section 5. Finally, Section 6 details the conclusions and potential future work.

2. Literature Review

According to the literature, detection of advertisement segments in television can be classified in two main classes: (1) detecting relevant features that are exclusive to advertisement or programming content, and (2) focused on the premise that an advertisement repeats multiple times through a broadcast. In the literature, these two methodologies are typically referred to as Feature-based and Repetition-based [2,3], respectively. The proposal of this paper aligns with the former, but differs by not making use of prior data regarding the channel, the stream (e.g., content at the start or at the end), or the corresponding content.

An early approach that incorporated both feature and repetition based methods was presented in [2]. This method incorporates two different detectors: The first detector analyses visual features such as black frame rate, motion, hard cuts, and fade rates from the advertisements; this process serves as an initial filter to find candidate advertisements. Then, a fingerprinting analysis is performed on the candidates to verify if the advertisement candidate can be found on an input database. This methodology is essentially based on the existence of black frames as delimiters to advertisement segments. Consequently, it failed to correctly detect advertisements in cases where black frames were not used. To tackle this problem, approaches based on detection of silences [4,5] were used for a joint analysis of video and audio data, but these also failed to detect advertisements if the broadcaster did not use black frames nor silences to delimit the ads.

Repetition-based methods generally require the existence of a database where information about the advertisements is stored. Sequence matching procedures are then applied to input streams in order to search the database and identify advertisements. The database must generally be initially provided or built through the analysis of streams. In [6], a multimodal approach for detecting advertisements in repurposed videos is presented. It uses a repetition-based method that employs an analysis of audio and video features to determine if a given segment is repeated and check if it is inside a database. The database is first populated with advertisements obtained from identifying repeated segments in monitored streams, and it is automatically updated by adding to the database new repeating segments. Presented results indicated that this technique can reliably detect advertisements. However, the requirement for an initial dataset of known and used advertisements impairs its usage in many scenarios. In [7], a system that identifies repeating content on streams through the exploitation of dimension reduction techniques applied to audio is presented. This system assumes no prior knowledge, as it learns how to identify and detect the repeating content. Other similar methods have been proposed [8,9], but a common limitation is that advertisements that appear a single time in a stream will not be detected. Recent approaches based on near duplicate video retrieval, such as that in [10], may also be applied to advertisement detection. However, again they suffer from the aforementioned problems.

Looking more closely to feature-based advertisement detection methods, the channel logo has been considered a discriminative feature and is widely used. This happened under the premise that television broadcasters exclusively transmit their respective logo when programming content is airing, removing or washing it out when advertisements are displayed. In [11], an edge map-based logo detector is presented. It obtains logo candidates by analyzing gradient images and processes them using a pre-trained SVM, which validates if the edge map belongs to a known logo. The results reported were promising, but the requirement for a pre-trained SVM that knows all possible logos is a problem when applying this method to unconstrained scenarios. Another method for detecting logos was presented in [12]. It divides the detection into two different workflows: one for detection and removal of opaque and semitransparent logos, and the other for animated logos. The first workflow estimates the position of the logo by computing an average gradient map over several frames. For the animated logos, the image is divided into a grid and, for each block, the period of change and its associated quality are estimated. These metrics are then used to identify the period of animation and location of the animated logo. However, the only results reported are related to the quality of the image after the logo removal, without any mention on the actual logo detection capabilities nor datasets used. This hinders the perception of how well the algorithm actually performs when applied exclusively to logo detection.

Histogram of Oriented Gradients (HOG) [13] has been used in many object detection problems, due to the quality of the extracted features, including application in logo detection problems. For instance, in [14] spatial information is coupled with HOG to describe the channel logos of a database in order to define a library of template logos. The logo detection is then accomplished by comparing the extracted features of each input frame with the logo database. A similar method based on the usage of color segmentation has been proposed in [15] that first applies a low-cost color segmentation to obtain candidate logo locations, that are then processed by an SVM trained with HOG features obtained from a logo dataset. However, a comparison with state-of-the-art methods is not reported nor does it present details of the dataset used. Note that the usage of color in the context of advertisement detection may not always be straightforward, as there are multiple color spaces and possible metrics [16] that may convey different information.

The difficult task of logo detection is made harder due to inter-class similarity and intra-class difference in the logo images [17]. Currently, deep learning has demonstrated impressive performances in object detection, with many object detection methods being proposed, including Faster R-CNN [18], YOLO [19], and SSD [20], but at the cost of large amounts of data. More recently, anchor-free detection methods [21,22] have achieved better performance compared to two- and one-stage anchor-based detection methods, but they also require extensive amounts of training data. Deep learning-based object detection methods have been used in logo detection, but without focusing on the case of broadcasting content. István et al. [17] trained a detector with logo and non-logo background to retrieve logo images. Su et al. [23,24] studied the effect of data augmentation by creating synthesized logo images for model learning. In Su et al. [25], the authors identified the most compatible training images with logo examples from a noisy image-level dataset to iteratively trained a model. Jain et al. [26] proposed the LogoNet architecture that includes a spatial attention module and hourglass-like feature extraction backbone, and while reported experiments show approximately 1.5% improvement in performance compared to state-of-the-art anchor-free detection network on FlickrLogos-32 dataset, the proposed approach requires knowledge of the possible logos and does not target broadcasting content.

A deep learning weakly supervised approach for logo detection in broadcasting is presented in [27]. The proposed system combines two different networks: a Region Proposal Network (RPN) and a Fast R-CNN network [18]. The RPN returns positive and negative sample proposals for logo location. However, it requires manual selection of the positive samples for the training. This network is then used to extract positive and negative samples of logos that are annotated and fed to the Fast RCNN to train the detector. Although this approach obtains good results, it requires manual annotation of the logos. This means that a fine-tuning step is required when applying this method to different channels that are not part of the initial training process.

A recent approach that does not require previous trained models is presented in [3]. It also follows the general premise that the presence of the channel logo is correlated to the content being a program and not an advertisement. The logo identification process is based on an analysis of digital on-screen graphics, which are watermark-like digital objects that are displayed on top of the transmitted content. It uses an unsupervised approach for detecting logo candidates on the image through an analysis of edge maps obtained from the frame corners. Edge maps are calculated at each frame and are inserted into a database depending on their characteristics. The reasoning between distinguishing general logos from channel logos is made through a series of thresholds that need to be fine-tuned to provide the best performance. Results show that the method is able to obtain interesting results on the author's testing dataset. However, the videos used in the testing are very short and contain manually introduced advertisements, which is not a good representative of real broadcasts. Furthermore, results are not compared with state-of-the art methods.

This analysis of state-of-the-art methods for advertisement detection shows that the general consensus is that the channel logo can be a good indicator for identifying programming content. The dual of this observation is that whenever the channel logo is absent, the content belongs to the advertisement. Even though it is a common trait in Europe, this concept is not followed by every television station worldwide. Methods for logo detection mostly follow a supervised approach, where it is assumed that a database of target channel logos can be either produced manually or given as input, which implies less flexibility or a heavy burden in data preparation. This shows that there is a need for a truly automatic procedure that self-learns the characteristics of the logo and automatically trains a classifier targeted for each processed stream.

3. Dataset Description

Despite the existence of advertisement detection proposals in the literature, the corresponding datasets are rarely available, mainly due to intellectual property and legal issues.

Therefore, for the purpose of evaluating our proposal, a dataset used in the state-of-the-art was augmented with new captures of recent television content broadcasted during 2019. The assembled dataset is comprised of European broadcast streams captured from Digital Terrestrial Television (DTV) broadcasts and cable operators (the sequences were capture for test and development purposes in the context of the R&D project MOG CLOUDSETUP - No17561, supported by Norte Portugal Regional Operational Programme (NORTE 2020)). It is mostly comprised of streams from general content channel containing different types of programs including news, talk shows, movies, and sports. The streams are long (mostly over 1 h) to incorporate a large number of possible situations, including different types of initial states (program, advertisement, mid-program, and mid-advertisement). Moreover, the dataset was manually annotated to delimit the different segments, to be used as ground truth as well to as associate different audiovisual features to the segments. A general overview of the captured videos is presented in Table 1.

Table 1. General overview of the captured streams.

Channel	Content	Provider	Duration	Ads Duration
RTP1a	Variety	DTV	6 h 52 min	0 h 23 min
RTP1b	Variety	DTV	1 h 10 min	0 h 11 min
RTP2	Variety	DTV	8 h 25 min	0 h 24 min
SIC	Variety	DTV	8 h 37 min	2 h 18 min
TVI	Variety	DTV	8 h 36 min	2 h 09 min
CMTV HD	Variety	Cable	0 h 42 min	0 h 06 min
Eurosport 1 HD	Sports	Cable	0 h 42 min	0 h 05 min
FOX HD	Series	Cable	1 h 35 min	0 h 24 min

The main audiovisual characteristics considered can be summarized as follows:

- Sign Language: the presence of a sign language interpretation;
- Banner/Crawler: the presence of a moving line of text during TV news programs or in other contexts;
- Channel Logo: the presence of the television channel logo;
- Subtitles: the existence of subtitles;
- Other Logos: Any other on screen logo that is different from the channel logo;
- Transition Silences: short pauses with no sound;
- Dialog: multiple speakers in sequence;
- Background Music: the presence of songs or jingles.

Table 2 summarizes the presence of these audiovisual characteristics in advertisements and programs, thus providing an indication of their discriminating capabilities. The results show that the channel logo is the visual feature that can be used with greater confidence in deciding if a frame belongs to a program or advertisement segment. Although other features have a more reduced presence in advertisements, such as sign language or subtitles, they are also less present in programs, resulting in a more balanced usage. The presence of black frames was also observed in our analysis. However, they occur in both types of content with seemingly no correlation to the content it divides. This illustrates the reason why their usage in advertisement detection scenarios has deprecated. Furthermore, note that other types of logo may be present in either programs or ads increasing the difficulty and complexity of the problem. Although the audio features appear in significant portions of the content, they appear both in program and advertisement content. This renders their usage more difficult, as a detailed analysis of the changes between their features in advertisement and program is required.

Table 2. Temporal distribution of the presence of the visual features.

Features	Program Content (%)	Advertisement Content (%)
Channel Logo	97.9%	3.1%
Crawler	36.2%	4.1%
Sign Language	14.7%	1.7%
Subtitles	34.9%	3.0%
Other Logos	40.5%	23.5%
Transition Silences	57.2%	86.4%
Dialog	92.4%	25.2%
Background Music	70.0%	93.3%

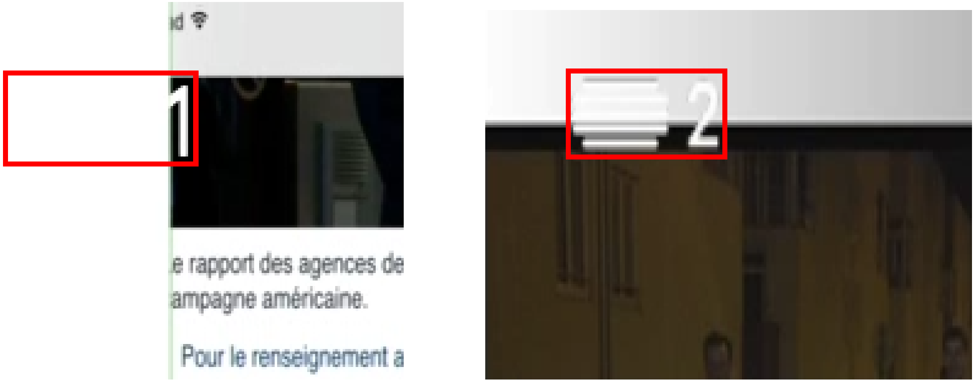
Based on the this analysis, as well as on the state-of-the-art, the current proposal exploits the channel logo as visual feature. Given the many different possible characteristics that channel logos can assume (e.g., transparencies, animations, position, and size), in Table 3 we provide further detail on the characteristics of the gathered dataset with regards to the channel logo. Although channel logos have some similar characteristics, significant differences are also noticeable, including size, shape, and color. Even in the same channel it is possible to observe the usage of different logos in distinct points in time or a periodic switching between logos. These many possible variations augment the difficulty of automatic logo detection. For instance, the RTP1a and RTP2 streams have a monochromatic logo that is absorbed or becomes partially invisible in multiple situations, as illustrated in Figure 1. As an additional example, let us consider the RTP1b and Eurosport streams. The first contains mostly news content, with an almost permanent crawler and other on screen objects, making it extremely difficult to just use temporal consistency information. Additionally, the channel logo varies between the generic RTP1 logo and a special 60 years anniversary logo through random transitions, as seen in Figure 2. The Eurosport stream contains mostly sports content and uses a transparent channel logo. Consequently, the logo blends with the background making its detection harder, as depicted in Figure 3.

The collected dataset also encompasses situations of self-advertising. These are very complex situations that hinder the frame classification process. The RTP2 and TVI streams have announcements of upcoming programs which contain a grayed version of the channel logo. Examples of these situations can be seen in Figure 4.

An important aspect that must be highlighted is that, even in the 8 h streams, there are multiple advertisements that appear only once. This indicates the unsuitability of using advertisement detectors based on finding repetitions. A closer look at the dataset shows that there are 978 individual segments with an average time per segment of approximately 26 s; the larger observed segment is 1 min and 10 s long. The advertisements always appear grouped into blocks with a varying number of segments and duration.

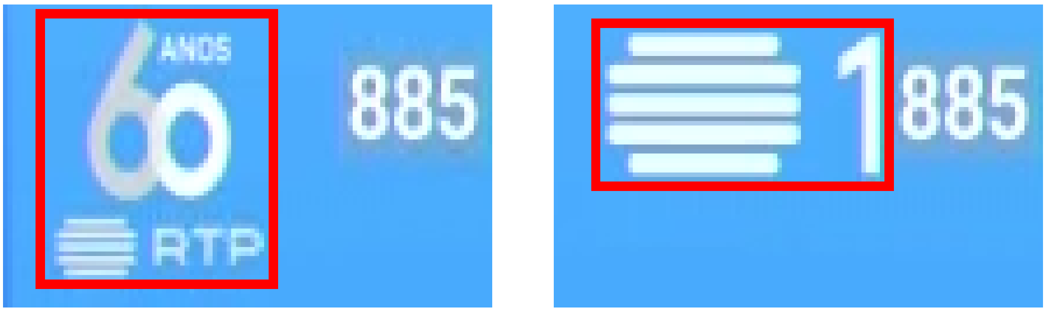
Table 3. Description of the general characteristics of the logos in the annotated dataset.

Logo	Channel	Stillness		Colored	Opacity	Position	Relative Size (% of the Screen)
		Shape	Texture				
	RTP1a	Static	Static	Monochromatic	Opaque	Top Left Corner	0.47%
	RTP1b	Changes	Changes	Monochromatic	Opaque	Top Left Corner	0.67% and 0.47%
	RTP2	Static	Static	Monochromatic	Opaque	Top Left Corner	0.45%
	SIC	Static	Changes	Colored	Opaque	Top Left Corner	0.42%
	TVI	Static	Changes	Colored	Opaque	Top Left Corner	0.50%
	Eurosport	Static	Static	Monochromatic	Transparent	Top Right Corner	0.48%
	FOX HD	Static	Static	Monochromatic	Opaque	Top Right Corner	0.48%
	CMTV	Static	Changes	Colored	Opaque	Top Left Corner	0.65%



(**a**) (**b**)

Figure 1. Example of a frame from RTP1a and RTP2 streams where the logo is only partially visible. In the left frame (**a**) the logo has only a small visible part and in the frame on the right (**b**) there is a larger visible portion, but blending with the background.



(**a**) (**b**)

Figure 2. Comparison between the two logos present on RTP1b stream. (**a**) RTP1b special 60 years logo. (**b**) General RTP1b logo in the immediate next frame.

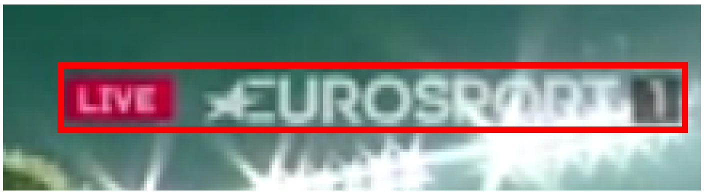
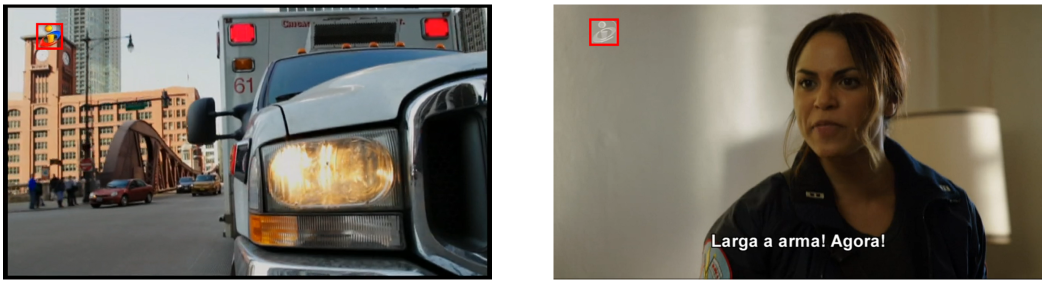


Figure 3. Eurosport logo blended with the background as seen in the final part of the logo.



(**a**) (**b**)

Figure 4. Comparison between the general TVI channel logo and the logo used to advertise an upcoming program. (**a**) Program scene where the typical colored version of the channel logo is visible in the top left corner. (**b**) Grayscale version of the logo on an advertisement of an upcoming program.

Given the differences between streams and even within the same stream, we defined three increasingly difficulty levels, based on visual complexity, to classify the streams and enable an objective assessment. The levels are the following:

- Level 1: Streams with opaque logos in a static position and background distinguishable.

- Level 2: Streams with logos in a static position, where the logo is often very similar to the background or indistinguishable.
- Level 3: Streams with varying or animated logos, where the logo is often similar or blends to the background.

Following the definition of the difficulty levels and the inherent characteristics of the streams, the following classification was made:

- Level 1: SIC, TVI, and FOX HD;
- Level 2: RTP1a, RTP2, and CMTV;
- Level 3: RTP1b and Eurosport.

To perform an objective comparison with state-of-the-art methods, the dataset presented in [3] was also used. It is composed of only three artificially generated streams using recordings of three different Portuguese channels and some repeated advertisements. The total duration of these streams is 2575 s and the general characteristics can be described as follows:

- Stream SICNot_BL: Recording of a news program of the SIC channel, with three sudden cuts to advertisement.
- Stream TVI_BL: Three different programs of the TVI channel, divided by two advertisement blocks containing the same advertisements.
- Stream RTP_BL: Similar structure as the SICNot_BL stream but with recordings of the RTP1 channel.

Based on the characteristics of these streams and on the criteria defined for the three difficulty levels, they were assigned to the lowest level of difficulty.

The classification of the collected set of streams into the defined difficulty levels resulted in an unbalanced distribution, where most of the content was classified with either difficulty level 1 or 2. Additionally, due to the single classification of long streams, some content with challenging details was absorbed into a unique classification. For a more detailed analysis and characterization of the streams, the dataset was divided into smaller segments, summarized in Table 4. This split in smaller streams was accomplished by applying the proposed difficulty levels to contiguous portions of the complete streams, without tampering with the advertisement blocks nor the programming content.

Table 4. Description of the substreams obtained from the original dataset.

Stream Name	Description	Difficulty	Length
eurosport_cut1	Snowboard content with multiple on-screen graphics and blended logo	3	00:19:00
eurosport_cut2	Football game. Scoreboard on top left corner	1	00:23:00
tvi_cut1	Program with self logo	2	00:44:00
tvi_cut2	Movies	1	05:00:00
tvi_cut3	News. Multiple on-screen graphics	2	01:42:00
tvi_cut4	Program with self logos	2	01:10:35
rtp1a_cut1	News. Multiple on-screen graphics	2	00:20:00
rtp1a_cut2	Live show with very bright scenes.	3	01:38:40
rtp1a_cut3	News. Multiple on-screen graphics	2	04:52:55
rtp2_cut1	Programs with just the channel logo.	1	04:08:29
rtp2_cut2	Movie with mostly white background.	3	01:40:21
rtp2_cut3	Program with other on-screen graphics.	1	01:47:01
rtp2_cut4	News. Multiple on-screen graphics	2	00:49:29
sic_cut1	Program with just the channel logo	1	00:46:00
sic_cut2	Program with program logo.	1	00:43:00
sic_cut3	Movie with movie logo	1	01:49:00
sic_cut4	Movie with movie logo	1	02:25:00
sic_cut5	News. Multiple on-screen graphics	2	01:43:00
sic_cut6	Program with program logo.	1	01:10:49
SICNot_BL [3]	Program with program logo.	1	00:04:33
TVI_BL [3]	Program with program logo.	1	00:11:48
RTP_BL [3]	Program with program logo.	1	00:02:49

4. Logo-Based Advertisement Detector

The dataset described in Section 3 encompasses channel logos with heterogeneous characteristics (e.g., form, aspect ratio, size, opacity, color, and animation) which is aligned with the state-of-the-art [14]. Note that there may be logos on screen that are not from the channel, as advertisements, and even certain programs, can have different/their own logo, as depicted in Figure 5. However, there are certain common characteristics that can be exploited. For example, channel logos are typically placed on a corner of the image and are temporally consistent. It may occur that the logo switches sides even during a program, but even in these cases, there is a constant stability of the position before and after the change in position.

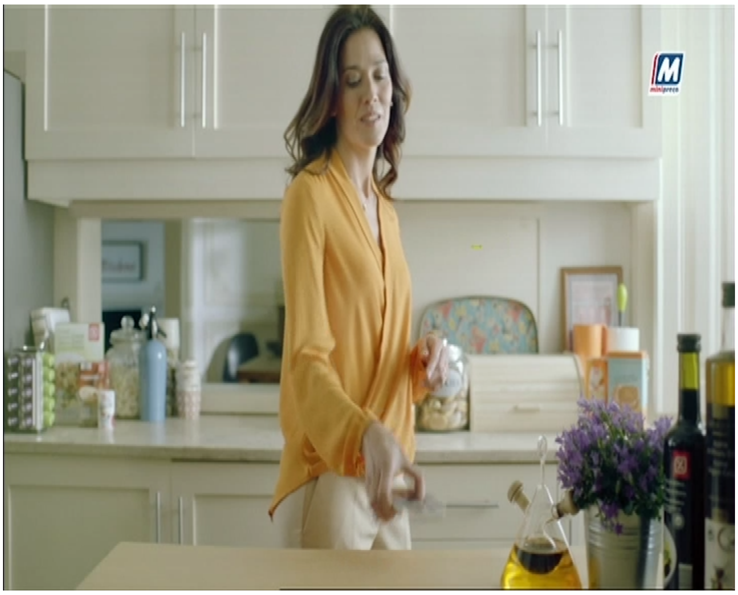


Figure 5. Advertisement with the brand's logo in top right corner.

Based on all previous observations and insights, a video-based advertisement detection methodology is defined focusing on the automatic identification and detection of the channel logo for detecting advertisement segments without using prior data about the channel, logos, or streams to be processed. A high-level overview of the proposed approach is presented in Figure 6.

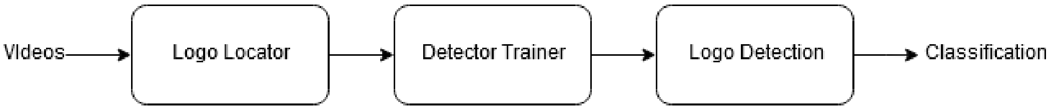


Figure 6. High-level conceptual view of the advertisement detection algorithm.

In the proposed architecture, each input video is first processed by the Logo Locator module, which is responsible for automatically identifying and locating possible locations of the channel logo using an unsupervised approach. The goal is to identify logo candidates as quickly as possible to minimize the delay introduced before conveying decisions on the frames and consequently on program/advertisement segments. A feature extraction methodology is then applied to each of the located regions to form a balanced training dataset of positive and negative samples of each logo candidate. Then, an automatic training procedure is applied to this data in order to obtain object detectors tailored to each of the identified logo candidates. The extraction of features and training of the logo detectors is done by the next module in the processing chain, the Detector Trainer. Finally, the Logo Detection module is responsible for applying the learned detectors in each frame of the input stream. Through this process, a given frame is assigned as belonging to a program if the detector(s) can find a logo on the image, or as advertisement otherwise.

The proposed architecture includes automatic feature extraction, labeling, and an online training procedure; thus, no human interaction is involved. Furthermore, no trained models or data initially required are stored, as each input video is considered independent and the complete training process is performed automatically, from the data extraction to the actual model training.

4.1. Logo Locator

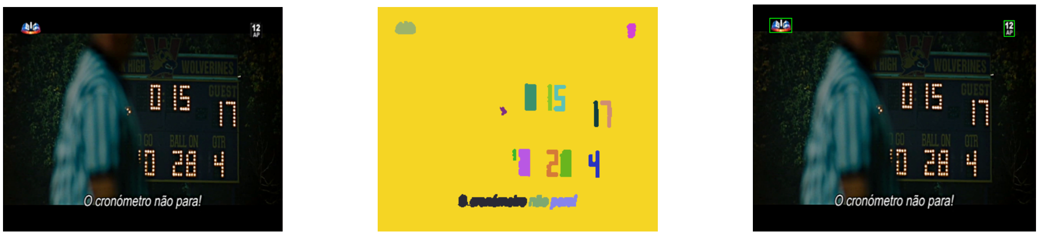
The proposed strategy does not use prior data or specific information about the input stream. Consequently, it is essential that the first module is able to automatically and robustly determine the location and group of pixels that correspond to the channel logo. As described in Section 3, the aspect and position of the channel logo does not usually change drastically throughout the stream, even in animated logo scenarios. With this insight, an image region is defined as a logo candidate region if it contains a group of pixels with similar colors and/or textures and shows spatio-temporal stability.

To obtain the logo candidate regions and, given that the channel logo is generally different from the background, an unsupervised region growing graph-based segmentation approach [28] is employed. This is a bottom-up segmentation algorithm that enables an hierarchical grouping of pixels into regions through the usage of a robust grouping strategy. It starts by considering that each pixel is a unique region and iteratively groups pixels whose neighbors are similar. For that, it compares the color, texture, size, and fit of regions and joins them into a single region if their similarity is within a similarity measure range. The last two characteristics used for this comparison force smaller regions not to merge with bigger regions, which is essential for the segmentation of the logo to not be absorbed by the background. The combination of color and texture through the usage of multiple color spaces also allows segmentation of regions even with monochromatic color. One important aspect that is provided from this hierarchical segmentation is that the scale of the objects segmented is not fixed, which allows the identification of logos with varying sizes.

As seen in the literature, a common trait associated with channel logos is that they are located on the corners of the image and not on the middle. This means that the search of the logo candidate regions can focus on the four regions surrounding the outermost corners of the image. Naturally, this not only reduces the amount of false positive detections, but also enables a faster detection. Therefore, a process similar to the one described in [3] was followed to partition the image into the four corners and evaluate the presence of the logo only on those regions. Note that, although the logos are assumed to be placed on the corners of the image, the proposed segmentation approach applies to the full image. This is a requirement, as we apply an hierarchical clustering agglomerating of the pixels, which enables the grouping of most of the background into a single region, resulting in a more controlled segmentation of individual regions. A common drawback in many segmentation procedures is that the resulting regions may contain segmentation errors. As such, it is necessary to remove these erroneous regions before a more detailed analysis of the segmentation is performed. To do this, we designed a simple filtering procedure that analyses the size and location of the candidate regions and removes the ones that are either too large or too small and not situated on the image corners. Based on our analysis of different channels and logos, we verify that their size is proportional to the area of the image. Therefore, we only consider acceptable logo regions if their area is between 0.35% and 1% of the image. These values were determined empirically and fixed for all experiments.

An underlying premise to the process is the need to be unsupervised, without prior data of the channel or input stream. Therefore, we apply the segmentation and filtering process in each individual frame. This results in temporally independent segmented regions. The next step consist in validating the logo candidate regions by analyzing their temporal consistency within a window of parameterized size. From the analysis of the dataset, advertisement segments account for just 26 s on average; we tested several window sizes around this value, and consider a region to be temporal consistent if it is segmented in the same location for at least 30 s.

To keep track and validate each logo candidate, the temporal consistency of the logos is modeled as a heat map with regions where the temperature reaches a certain value being considered as good logo candidates. To account for missed detections that may belong to segmentation clutter, two different processes are employed: heating, which is responsible to increase temperature given detection candidates, and annealing, that is applied to a region where no detection was found. For every frame and for every filtered detection, we increase the temperature on the corresponding region in the heat map by appending a circular mask that is centered on the middle point of the detection with diameter corresponding to the diagonal length of the region's bounding box. As static logos may be segmented differently according to their surroundings, resulting in (slightly) different regions in subsequent frames, this process enables a steady increase of the temperature in regions where constant detections are obtained. Some logo candidates may be derived from segmentation errors or even logos of advertisements that are present for a limited period, requiring the annealing of regions where no detection can be found. As a result, the temperature value of regions with consistent detections will be considerably superior to the remaining space. As this procedure is applied to each frame, it is necessary to specify a stopping criteria. Given our definition of temporal consistency, we specify the stopping criteria for the temperature as $T_{max} = 30 \times fps$, where fps represents the frame rate of the input video. To allow for multiple possible candidates, as some channels may have more than one logo, we also select as logo candidates all the regions where the temperature reaches up to 90% of T_{max}. An illustration of this procedure is presented in Figure 7. In Figure 7a, we have an unprocessed input frame; in Figure 7b, we have the resulting graph segmentation of the input region. As can be seen in the upper corners of the image, the two possible logo regions are clearly identified. Finally, in Figure 7c we have the extracted region of the candidate logos that were validated with temporal consistency.



(**a**) (**b**) (**c**)

Figure 7. Frame 939 from the sic_cut3 stream, with corresponding graph-based segmentation. (**a**) Input Frame. (**b**) Graph-based Segmentation. (**c**) Selection of logo candidates from the graph segmentation after spatio-temporal stability validation.

Whenever a logo candidate is found, the previous detections are re-analyzed, and patches that have a center location in the vicinity of the maximum temperature points are extracted. Thus, positive samples associated to each logo candidate are selected. To provide negative samples of the logo, we also extract random patches from the images. This information is then passed to the Logo Training module.

4.2. Logo Training and Detection

The input of the Logo Trainer module consists of the visual data extracted from the regions corresponding to the validated logo candidates, along with negative samples (random samples of images pages not overlapping with the candidate regions), which will be used to train an object detector for each candidate. From our experiments, we observed that a sample of 500 positive and negative samples of each region was sufficient to train a reliable detector, without significantly hindering the process time of the entire stream. These samples are extracted from the frames where the Logo Locator identified temporal and spatial consistent regions.

Different techniques were analyzed for logo detection with the final choice falling on the pair of HOG features [13] and SVM training [29]. Even though there is a tendency towards deep learning methodologies such as FasterRCNN for object detection, these require significant amounts of training data—a requirement that may not be compatible with the target scenario. Additionally, HOG+SVM has proved to achieve good results in classification problems, even with small amount of training data [30]. As such, they are a good combination to use in scenarios where small amounts of training data are available, such as the ones being targeted where we want the processing to be as causal as possible to decrease delays. Furthermore, note that the hardware available in many scenarios has some restrictions; this hinders dramatically the usability of deep learning approaches as, in addition to large amount of annotated data, the training process requires dedicated GPUs in order to reach good results in reasonable time.

To overcome situations where the logo is not fully and clearly visible, for example, due to the background (Figure 1), a set of detectors are trained to find parts of a logo. More specifically, the bounding box associated with each logo candidate is divided into 12 overlapping sub-regions, as illustrated in Figure 8. Each partial detector is trained only to the information extracted from the corresponding region. The output of the individual detector is used to decide on the presence of the logo (and consequently classification of the frame as program) through a majority voting.

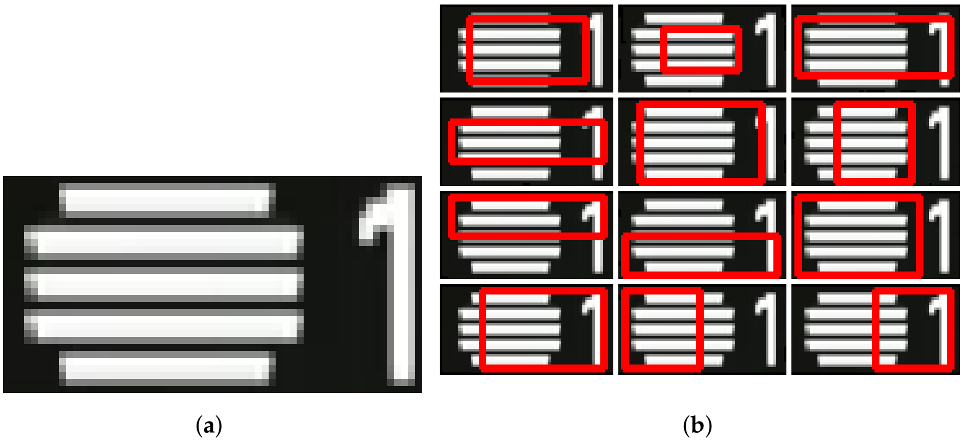

(**a**) (**b**)

Figure 8. Example of a detected logo candidate and associated additional subregions. The reader is encouraged to visualize this image in color to better see the subregions. (**a**) Region of the validated logo candidate. (**b**) The 12 additional subregions inprinted on the base logo.

For each candidate logo and associated subregions, we extract the HOG features from both positive and negative samples and perform a 5-fold cross-validation on the data. We then select the models that achieved better performance on the cross validation for each logo region and subregion. The SVM used in the training procedure is a Linear SVM with parameters $C = 5$ and $\epsilon = 0.001$. Both the selection of the parameters, as well as the kernel of the SVM were obtained by applying a grid search and selecting the combination that provided the overall best performance, which was then used for all experiments.

After the Detector Trainer finishes the training and selection of the logo detectors for each region and subregion, it then forwards the trained models for the Logo Detection module, which processes the remaining frames of the input stream providing a binary classification on the presence of the logo in each frame. Given the need to first locate and validate logo candidate regions, an unpredictable amount of frames at the beginning of the stream may be discarded. Therefore, the Logo Detection module applies the trained detectors over the start of the stream (excluding frames used in the training) to account for missed detections during the logo location and training phases.

5. Evaluation and Results

To objectively evaluate the proposed method, widely accepted metrics and approaches in the literature were employed. Given that the overall detector assigns a binary result for each processed frame, indicating if it either belongs to an advertisement or to a program, a binary classification evaluation was applied through the calculation of the Precision (Prec), Recall (also known as true positive rate, TPR), F1-Score (F1), and Accuracy (Acc) metrics [31]. For this, the following measures were used:

- True Positive (TP): Algorithm assigns the tag advertisement to a frame that belongs to an advertisement.
- False Positive (FN): Algorithm assigns the tag program to a frame that belongs to an advertisement.
- False Negative (FP): Algorithm assigns the tag advertisement to a frame that belongs to a program.
- True Negative (TN): Algorithm assigns the tag program to a frame that belongs in fact to a program.

Table 5 summarizes the results in terms of precision, recall, F1, and accuracy for the full streams of the datasets.

Table 5. Results obtained for the annotated dataset, analyzing the advertisement detections.

Channel	Prec	TPR	F1	Acc
RTP1a	29.10%	95.65%	44.62%	83.30%
RTP1b	45.18%	97.31%	61.71%	81.75%
RTP2	20.04%	54.92%	29.36%	82.19%
SIC	97.25%	87.46%	92.10%	95.97%
TVI	82.50%	98.86%	89.94%	94.47%
CMTV	46.57%	99.99%	63.54%	83.66%
Eurosport	58.57%	100.00%	73.87%	86.38%
Fox HD	99.12%	100.00%	99.56%	99.78%

The lowest results depicted in Table 5 can be easily explained due to abnormal occurrences in the streams, such as logo absorption with the background for very long periods of time, as illustrated in Figure 9. In this case, the logo was absorbed by the background for ~23% of the whole stream. This results in erroneous classifications of frames as being advertisements when in fact they belong to programs with no visible or perceived logo. Overall, results convey a significant performance with the proposed method being able to adapt to different channels and self-learn how to detect the channel logo and subsequently distinguish advertisements and program content. Most of the results show high accuracy with values above 80%.

Figures 10 and 11 provide a visual perspective on the results for the RTP2 and SIC streams. These depict, in a vertical stack, the plotting of the errors, the annotations, and obtained results. The blue bars indicate programming content and the yellow bars advertisements. The abnormal occurrences in the RTP2 are also visible in the temporal representation of the classifications illustrated in Figure 10. As can be seen, the amount of advertisement content in the stream is vastly inferior to the amount of programming content. If we focus on the first highlighted region, around frames 200 k, we see that some advertisements were not detected. This occurred due to self-advertisement in which the channel logo is present on the advertisements. Naturally, this occurrence impacts significantly on the recall rate. Another aspect is illustrated on the second region, between frames 380 k and 510 k (a considerable portion of the stream) where program was classified as advertisement due to logo absorption. An example of results in a common stream is depicted in Figure 11, with only some isolated frame level errors and visible significant performance of the proposed method.

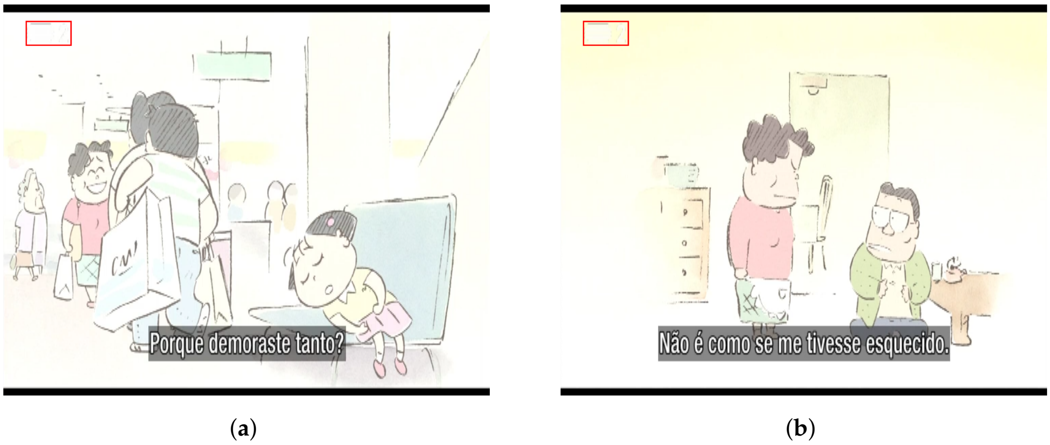


Figure 9. Example of frames from the RTP2 stream where the detector failed to find the logo. In the left frame (**a**) the logo is imperceptible, even for humans. The frame on the right (**b**), show the logo just slightly perceptible.

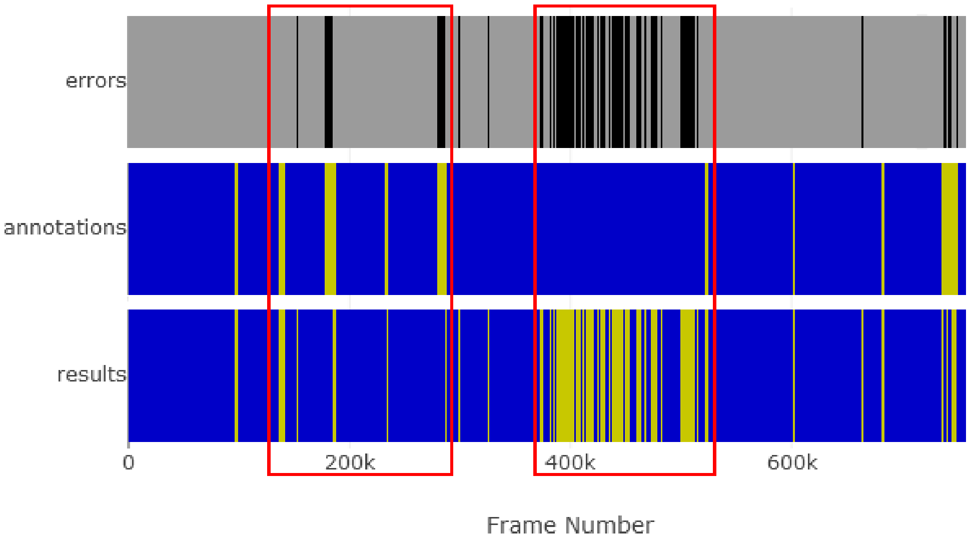


Figure 10. Comparison of the results obtained for the RTP2 stream with the annotated data, illustrated as black bars the classification errors.

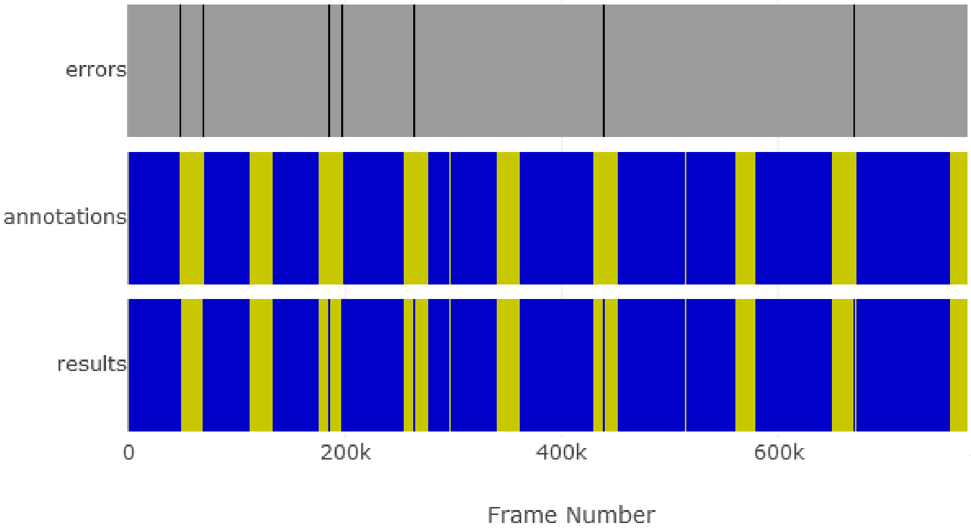


Figure 11. Comparison of the results obtained for the SIC stream with the annotated data, illustrated as black bars the classification errors.

A different perspective is provided in Table 6 with the results for all the sub-segments aggregated by difficulty level (see Table 4). These demonstrate very high performance in the lower difficult streams (1 and 2), but also in the highest level (3) ones. The results obtained for the easier streams indicate that the algorithm performs very well in cases where the channel logo is clearly distinguishable from the background content. Moreover, note that the performance for levels 2 and 3 is very close (~90%). Streams with high quantity of visual clutter and with monochromatic logos (very similar to the programs being broadcasted) pose some obstacles to the proposed approach. This happens due to the logo absorption problem which is more predominant in monochromatic logos. Even manual observation of the logo poses severe difficulties since no perceived visual difference between the logo or background is noticeable. Another identified difficulty occurs when streams do not follow the base hypothesis in which programming content is displayed without logo and self advertisements are presented with the channel logo.

Table 6. Average results obtained for the sub-streams of the dataset, stratified by the difficulty level.

Difficulty	Precision	Recall	F1	Accuracy
1	89.3%	86.3%	85.6%	95.7%
2	79.2%	93.6%	85.6%	90.3%
3	48.7%	66.6%	55.7%	88.0%

For a more detailed and objective comparison with the state-of-the-art, the testing dataset presented in [3] was also manually annotated and processed with the proposed method. The results obtained and the baseline reported in [3] are presented in Table 7. To the best of our knowledge, the method presented in [3] is the most similar to our approach, as it follows the same principles of not requiring manual data labeling for processing input streams and being able to adapt to streams with different logos without manual adjustments or parameter corrections. From this comparison, it is clear that our proposal outperforms the state-of-the-art, obtaining almost ideal detections for this dataset. Note that the three streams that composed the dataset were assigned the lowest difficulty level, i.e., they are less challenging than most of the segments in the full dataset. Moreover, unlike the remaining dataset, the three streams start with the logo present, which simplifies the problem.

Table 7. Comparison of the results obtained using the proposed method and the advertisement detection method proposed in [3].

Method	Prec	TPR	F1	Acc
		SICNot_BL stream		
Gomes et al. [3]	100%	89.4%	94.4%	98.1%
Our Proposal	99.9%	99.8%	99.9%	99.9%
		TVI_BL stream		
Gomes et al. [3]	94.6%	93.2%	93.9%	95.1%
Our Proposal	100%	100%	100%	100%
		RTP_BL stream		
Gomes et al. [3]	100%	94.2%	97%	97.8%
Our Proposal	99.7%	99.8%	99.8%	99.8%

6. Conclusions and Discussion

This paper proposes a self-learning and automatic detection method for detecting television advertisement without using any prior data about the streams. The underlying assumption is that advertisement frames do not contain the channel logo; this is supported both by the state-of-the-art and the analysis performed over the collected dataset that demonstrated the high discriminating power of this visual feature.

The proposed solution automatically identifies candidate regions where the channel logo may be located and trains an object detector and partial sub-detectors using HOG features extracted from the logo location. The use of the partial detectors enables identification of the regions even in situations where the logo is only partially visible. The process is completely automatic as the extraction of training data is also controlled by the algorithm.

For an objective assessment of the algorithm, a dataset comprised of multiple Portuguese and international channels was collected and annotated. Additionally, the streams were broken down in smaller segments of similar characteristics and assigned one of three difficulty levels. For a comparison with the state-of-the-art, the current proposal was also assessed on the testing dataset of a state-of-the-art advertisement detection method.

Even though the advertisement detection may be impaired by abnormal situations, such as the channel logo becoming invisible due to background absorption or because of self-advertising, results show that the proposed method surpasses the state-of-the-art, achieving significant performance on multiple types of streams, including those of greater complexity.

The current proposal may benefit of additional work at the level of postprocessing to increase detail in relevant regions and circumvent some of the missed detections due to the absorption of the logo by the background. Besides using the channel logo, performance may also be improved by incorporating the detection of other visual features, even if these have lower discriminating power, and specially by also using audio in a multimodal approach.

Author Contributions: Conceptualization, A.P., P.V. and P.C.; methodology, A.P., P.V. and P.C.; software, A.P.; validation, A.P. and P.C.; formal analysis, A.P. and P.C.; investigation, A.P. and P.C.; resources, A.P. and P.C.; data curation, A.P.; writing—original draft preparation, A.P. and P.C.; writing—review and editing, A.P., P.V. and P.C.; visualization, A.P., P.V. and P.C.; supervision, P.V. and P.C.; project administration, P.C.; funding acquisition, P.V. and P.C. All authors have read and agreed to the published version of the manuscript.

Funding: The work presented was financed by National Funds through the Portuguese funding agency, FCT—Fundação para a Ciência e a Tecnologia, within project UIDB/50014/2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Restrictions apply to the availability of the data. Legal constraints currently prevent the public sharing of the datasets.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results'.

References

1. European Parliament. Directive 2010/13/EU of the European Parliament and of the Council of 10 March 2010 on the Coordination of Certain Provisions Laid down by Law, Regulation or Administrative Action in Member States Concerning the Provision of Audiovisual Media Services. 2010. Available online: http://data.europa.eu/eli/dir/2010/13/oj (accessed on 10 June 2021).
2. Lienhart, R.; Kuhmunch, C.; Effelsberg, W. On the detection and recognition of television commercials. In Proceedings of the IEEE international Conference on Multimedia Computing and Systems, Ottawa, ON, Canada, 3–6 June 1997; pp. 509–516.
3. Gomes, A.; Queluz, M.P.; Pereira, F. Automatic detection of TV commercial blocks: A new approach based on digital on-screen graphics classification. In Proceedings of the 2017 11th International Conference on Signal Processing and Communication Systems (ICSPCS), Surfers Paradise, Australia, 13–15 December 2017; pp. 1–6.
4. Marlow, S.; Sadlier, D.A.; McGeough, K.; O'Connor, N.E.; Murphy, N. Audio and video processing for automatic TV advertisement detection. In Proceedings of the Irish Signals and Systems Conference (ISSC), Maynooth, Ireland, 26–27 June 2001; pp. 25–27.
5. Sadlier, D.A.; Marlow, S.; O'Connor, N.; Murphy, N. Automatic TV advertisement detection from MPEG bitstream. *Pattern Recognit.* **2002**, *35*, 2719–2726. [CrossRef]
6. Covell, M.; Baluja, S.; Fink, M. Advertisement detection and replacement using acoustic and visual repetition. In Proceedings of the 2006 IEEE Workshop on Multimedia Signal Processing, Victoria, BC, Canada, 3–6 October 2006; pp. 461–466.
7. Herley, C. ARGOS: Automatically extracting repeating objects from multimedia streams. *IEEE Trans. Multimed.* **2006**, *8*, 115–129. [CrossRef]

8. Li, Y.; Zhang, D.; Zhou, X.; Jin, J.S. A confidence based recognition system for TV commercial extraction. In Proceedings of the Nineteenth Conference on Australasian Database—Volume 75; Australian Computer Society, Inc.: Darlinghurst, NSW, Australia, 2008; pp. 57–64.
9. Zou, F.; Chen, Y.; Song, J.; Zhou, K.; Yang, Y.; Sebe, N. Compact image fingerprint via multiple kernel hashing. *IEEE Trans. Multimed.* **2015**, *17*, 1006–1018. [CrossRef]
10. Hao, Y.; Mu, T.; Hong, R.; Wang, M.; An, N.; Goulermas, J.Y. Stochastic Multiview Hashing for Large-Scale Near-Duplicate Video Retrieval. *IEEE Trans. Multimed.* **2017**, *19*, 1–14. [CrossRef]
11. Özay, N.; Sankur, B. Automatic TV logo detection and classification in broadcast videos. In Proceedings of the 2009 17th European Signal Processing Conference, Glasgow, UK, 24–28 August 2009; pp. 839–843.
12. Mikhail, E.; Vatolin, D. Automatic logo removal for semitransparent and animated logos. In Proceedings of the GraphiCon 2011, Moscow, Russia, 26–30 September 2011; pp. 26–30.
13. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893.
14. Zhang, X.; Zhang, D.; Liu, F.; Zhang, Y.; Liu, Y.; Li, J. Spatial HOG based TV logo detection. In Proceedings of the Fifth International Conference on Internet Multimedia Computing and Service, Huangshan, China, 17–19 August 2013; pp. 76–81.
15. Ye, F.; Zhang, C.; Zhang, Y.; Ma, C. Real-time TV logo detection based on color and HOG features. In Proceedings of the 2013 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), London, UK, 5–7 June 2013; pp. 1–5.
16. Pereira, A.; Carvalho, P.; Coelho, G.; Côrte-Real, L. Efficient CIEDE2000-based Color Similarity Decision for Computer Vision. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *30*, 2141–2154. [CrossRef]
17. Fehérvári, I.; Appalaraju, S. Scalable Logo Recognition Using Proxies. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 7–11 January 2019; pp. 715–725.
18. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
19. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
20. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 21–37.
21. Law, H.; Deng, J. CornerNet: Detecting Objects as Paired Keypoints. *Int. J. Comput. Vis.* **2020**, *128*, 642–656. [CrossRef]
22. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint Triplets for Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–3 November 2019; pp. 6568–6577.
23. Su, H.; Zhu, X.; Gong, S. Deep Learning Logo Detection with Data Expansion by Synthesising Context. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 24–31 March 2017; pp. 530–539.
24. Su, H.; Zhu, X.; Gong, S. Open Logo Detection Challenge. In Proceedings of the British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, 3–6 September 2018; p. 16.
25. Su, H.; Gong, S.; Zhu, X. WebLogo-2M: Scalable Logo Detection by Deep Learning from the Web. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 22–29 October 2017; pp. 270–279.
26. Jain, R.K.; Watasue, T.; Nakagawa, T.; Sato, T.; Iwamoto, Y.; Ruan, X.; Chen, Y.W. LogoNet: Layer-Aggregated Attention CenterNet for Logo Detection. In Proceedings of the 2021 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 10–12 January 2021; pp. 1–6.
27. Zhang, Y.; Cao, X.; Wu, D.; Li, T. Weakly-supervised TV logo detection. In Proceedings of the 2017 32nd Youth Academic Annual Conference of Chinese Association of Automation (YAC), Hefei, China, 19–21 May 2017; pp. 1031–1036.
28. Uijlings, J.R.; Van De Sande, K.E.; Gevers, T.; Smeulders, A.W. Selective search for object recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [CrossRef]
29. Osuna, E.; Freund, R.; Girosit, F. Training support vector machines: An application to face detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 130–136.
30. Bilal, M.; Hanif, M.S. Benchmark Revision for HOG-SVM Pedestrian Detector Through Reinvigorated Training and Evaluation Methodologies. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 1277–1287. [CrossRef]
31. Tharwat, A. Classification assessment methods. *Appl. Comput. Informatics* **2018**, *17*, 168–192. [CrossRef]