DEEP-AD: A Multimodal Temporal Video Segmentation Framework for Online Video Advertising

RUXANDRA TAPU\textsuperscript{1,2}, (Member, IEEE), BOGDAN MOCANU\textsuperscript{1,2}, (Member, IEEE), AND TITUS ZAHARIA\textsuperscript{1}, (Member, IEEE)

\textsuperscript{1}Laboratoire SAMOV AR, ARTEMIS Department, Institut Polytechnique de Paris, Télécom SudParis, 91000 Évry, France
\textsuperscript{2}Telecommunication Department, Faculty of ETTI, University Politehnica of Bucharest, 060042 Bucharest, Romania

Corresponding author: Ruxandra Tapu (ruxandra.tapu@telecom-sudparis.eu)

This work was supported by the framework of the joint laboratory Artificial Intelligence for Television (AITV) established between Télécom SudParis and France Télévisions.

ABSTRACT In this paper, we introduce the DEEP-AD framework, a multimodal advertisement insertion system dedicated to online video platforms. The framework is designed from the viewer’s perspective, in terms of commercial contextual relevance and degree of intrusiveness. The main contribution of the paper concerns a novel multimodal temporal video segmentation algorithm into scenes/stories, which makes it possible to determine automatically the temporal instants that are the most appropriate for inserting advertisement clips. The proposed algorithm exploits various deep convolutional neural networks, involved at several stages. The video stream is first divided into shots based on a graph partition method. The video shots are then clustered into scenes/story units with the help of an agglomerative clustering methodology taking as input visual, audio and semantic features. Furthermore, in order to facilitate the user’s access to multimedia documents a novel thumbnail extraction method is proposed based on both semantic representativeness and visual quality information. Finally, the optimal advertisement insertion points are determined based on the ads temporal distribution, commercial diversity and degree of intrusiveness. The experimental results, carried out on a large dataset of more than 30 videos, taken from the French National Television and US TV series validate the proposed methodology with average accuracy and recognition rates superior to 88%. Moreover, when compared with other state of the art methods, the proposed temporal video segmentation yields gains of more than 6% in precision and recall rates.

INDEX TERMS Multimodal temporal video segmentation, thumbnail extraction from video scenes, commercial advertisement insertion based on semantic criterions, deep convolutional neural networks.

I. INTRODUCTION

In the recent years, with the rapid development of the near-ubiquitous broadband Internet access, live video streaming has become widely popular. In the statistics published by the ComScore organization\cite{1} it is shown that more than 187 million Americans watch over 48 billion online content videos every month.

Motivated by the huge potential in terms of business opportunities, the online video advertising market presents a rapid and sustained development. The key objective of online video advertising is to introduce a relevant ad, appealing the interest of the target consumers at a suitable moment while avoiding being intrusive.

In practice, a wide variety of ads insertion schemes have been proposed, including: overlay ads, pre-, mid- or post-roll commercial insertion. However, most free video streaming platforms display the advertisement either at the beginning or at the end of the video. In this way, the users are forced to view it, but in most of the cases they ignore it because of its irrelevance.

The TV broadcasters tackle the issue in a different manner and place the ads at a fixed moment in time. This strategy is today widely spread all over the world. Its main drawback comes from its highly invasive character: the user can be disturbed by the interruptions that occur in inappropriate moments and thus becomes less receptive to the commercial message transported by the ad. Another solution concerns the overlaid commercial systems\cite{2}, [3]. In this case, the ads are placed on a fixed location (e.g., right bottom corner)
regardless of the video content. The problem with such an approach comes from the fact that useful, relevant information in the video content can be occluded.

We claim that, in order to both increase the user viewing experience and make it receptive to advertisement, it is necessary to develop a methodology that is able to insert ads that are: (1) correlated with the video content and (2) inserted at temporal locations dynamically established based on content discontinuity i.e., the end/start points of action plots. In this way, the user can understand why a given ad is proposed at a given moment and thus increase the degrees of acceptability and receptiveness. The associated technical challenges are twofold. First, we have to make sure to dispose of a highly precise video segmentation into semantically pertinent scenes/story units. Secondly, performant matching procedures are required in order to realize the appropriate alignment between content and ads.

In this paper, we introduce a novel advertising insertion methodology dedicated to online videos, so-called DEEP-AD, which notably aims at ensuring the ads contextual relevance, while minimizing its degree of intrusiveness. The proposed solution is able to dynamically establish the most appropriate temporal locations for the ads insertion, based on the analysis of visual content discontinuities. In addition, it makes it possible to match the content of the selected ad with the considered video segment. Moreover, the approach ensures that the ads insertion points are distributed as uniformly as possible along the video timeline.

The DEEP-AD framework, presented in Fig. 1, is based on computer vision algorithms and exploits multiple deep convolutional neural networks (CNNs) in order to deploy a sequence of operations including: shot boundary detection, face detection and tracking, person re-identification, object detection/recognition, video shots classification based on filming strategy, indoor/outdoor location identification, thumbnail extraction and relevant ads insertion. The main contributions of the paper can be summarized as follows:

1. An end-to-end approach for video temporal segmentation into shots and scenes, applicable to a large variety of video genders. The experimental results obtained (cf. Section IV) demonstrate that the proposed method is the most accurate, robust and generic video structuring solution;
2. A multi-modal scene boundary detection algorithm. Our proposal connects visual, audio and semantic cues extracted from deep neural networks architectures in order to optimally generate clusters of shots that are semantically connected. Compared with other state of the art techniques [4], [5], our approach is able to: differentiate between adjacent scenes containing the same actors, cluster singular shots (i.e., shots not satisfying the visual or audio similarity constraints) to the corresponding scene or create relevant, single-shot scenes;
3. An iconic, representative frame selection technique, which makes it possible to determine a relevant thumbnail, respecting both visual and semantic criteria for each video scene;
4. An automatic similarity assessment procedure, able to determine semantically relevant matches between advertisements and corresponding video segments.

The rest of the paper is organized as follows: in Section II we review the state-of-the-art approaches dedicated to online video advertising. Section III introduces the proposed methodology and describes the main steps involved: shot boundary detection, face detection and tracking, person re-identification, key-frames description based on visual features, object detection/recognition, video shots classification based on filming strategy, indoor/outdoor location identification, thumbnail extraction and relevant ads insertion. Section IV presents the experimental results obtained on a large set of videos. We show that it is possible to obtain high precision and recall rates for the temporal video segmentation on very challenging video sequences. Finally, Section V concludes the paper and opens some directions of future work and development.

II. RELATED WORK

In recent years, significant research work has been dedicated to the issue of online video advertising. Most techniques can be classified into two major families: (1) methods based on the analysis of the user’s behavior and (2) contextual video advertising methods, which aim at determining relationships between proposed ad and current, under view video content.

A. ADVERTISING INSERTION BASED ON USER BEHAVIOR

The ads insertion methods based on user behavior relay on marketing and psychological studies that aim at constructing the user profile depending on the associated searching, browsing and clicking history over the Internet. In [6], authors propose inferring the user’s interest and preferences from the viewing history. Then, based on a statistical analysis over a
group of individuals presenting the same behavior, a set of relevant advertisements is developed and pushed to the whole group of users.

In [7], the consumers’ attitudes and perceptions over different ads are measured with respect to multiple criteria, including: entertainment, informativeness, credibility and interaction. Authors point out that advertisement according to user interest can improve the user quality of experience and perception. The study was extended in [8], where it is shown that methods retrieving commercials based on user behavior have a higher impact over the consumer.

In [9], authors predict the popularity of a live video game section by analyzing the number of viewers of the Twitch stream in its early stages. While in [10], broadcasters and views of the Twitch platform are interviewed in order to determine the personalities that influence the viewers to stay on a specific channel.

However, the ads insertion systems based on the user behavior are complex to design and hard to mathematically model. The marketing and psychological studies are costly to conduct and do not scale for a large groups of individuals. In addition, the user behavior can change over time and be affected by other factors such as: mood or health.

B. CONTEXTUAL VIDEO ADVERTISING

Within this context, let us first mention the textual video advertising approaches, which relay on matching online textual content (which usually is available on the web pages that are hosting the videos) with the keywords associated to each ad, in order to determine a correlation between the target video and the advertisement.

One of the most popular frameworks, so-called AdSense [11], proposes to insert the commercials by matching the textual content of the web page in which the video is embedded to the keywords extracted from the advertisement. In a similar manner, the AdWords [12] system attempts to match the user keywords provided in the search query to the keywords of ads. Moreover, in [13], authors propose developing ten different types of matching strategies, within the context of a dynamic programming approach, in order to improve the performances of the keyword matching strategies.

Even though such approaches are today highly popular, the textual information associated to a video element is currently by far insufficient and fails to provide a precise description of the video content in its whole.

In addition, the usual problems of textual annotation subjectivity can lead to mislabeling. The performance of the online advertisement system is thus directly influenced by the matching strategy and limited by the low relevance between ad and target video.

Recent researches propose the so-called fixed point advertisement strategy. In this case, the ads are inserted at four places: in the beginning (pre-roll), at the end (post-roll), in the middle (mid-roll) or overlaid (the ad covers a part of the video stream -- bottom area).

However, in order to find the optimal insertion position based on video content analysis, most works existent in the state of the art follow a unified rule of placing the ad at the level of scene/story boundaries.

The video temporal segmentation into scenes/stories is an essential pre-processing task in a wide range of applications such as: video indexing, multimedia retrieval/browsing and classification. In this context, many systems dedicated to commercial video segmentation have been introduced in the state of the art. Various methods [14], [15] use low level visual features such as: the HSV color histograms or SURF interest points [16], globalized with a Bag of Visual Words [17] representation for video scene segmentation. Other techniques [18], [19] propose to use audio events in order to detect scenes boundaries. More recently, multimodal approaches exploiting both audio and visual features [20]–[22] have been introduced, in order to enhance the efficiency of the video temporal segmentation process.

A different area of research has been focused on the egocentric video structuring into coherent segments. In [23], a system able to automatically organize egocentric videos into chapters based on the visual content is proposed. In [24], the authors propose to use 3D-CNN for long-term activity recognition in egocentric videos to temporally segment long and unstructured videos. While, in [25] a framework able to analyze the egocentric video acquired by users and to segment it into coherent shots related to the specified personal locations is proposed.

The following part of this section is focused on temporal video segmentation systems specifically designed and tuned in the context of ads insertion. In particular, the vADeo system introduced in [26] determines the advertisement insertion time based on the detected scene boundaries. However, the correlation between the inserted ad and the video content is not taken into consideration.

In order to determine the ad insertion points, the VideoSense system introduced in [27] exploits several criteria, including a contrast measure between frames of the video stream, the variation of the motion vectors and the inconsistencies determined in the audio stream. The candidate locations are estimated with the help of a nonlinear integer programming approach that takes as input the frames degree of discontinuity and level of attractiveness.

A contextual video advertisement system denoted by AdOn is introduced in [28]. The method determines the advertisement insertion time by computing the shot duration and the global magnitude of the associated motion vectors.

The VideoAder system, dedicated to Web video advertisement, is introduced in [29]. Content-based object retrieval techniques are here used in order to identify the correlation between an ad and a potential embedding position in the video stream. Then, the ad insertion task is formulated as an optimization problem that aims at maximizing the total money incomes of the system. A similar method designed to increase the advertisement revenues proposes to include publishing bid in the multimedia content [30].
Recently, in [31] a system that combines the online shopping experience with online video advertising is introduced. The framework proposes to redirect viewers to relevant on-line shopping web sites based on the video content.

With the help of a deep CNN architecture, an object-level video advertising approach is proposed in [4]. The method is dedicated to human clothing advertising and aims at minimizing the viewer degree of intrusiveness when ads are inserted. Finally, a content targeting video advertisement system is introduced in [5]. The ads insertion points are estimated by analyzing the video frames situated in the near vicinity of scene boundaries.

In a general manner, the state-of-the-art analysis highlights that the existing online video advertisement systems based on video content analysis employ only perceptual cues such as color features, contrast measures or motion vectors in order to determine the story boundaries. However, such approaches fail to deal with complex videos where adjacent scenes may contain similar visual patterns developed in different locations and contexts.

In addition, the scene detection based solely on the visual information fails to group all shots to the corresponding scene. Consequently, the advertisement risks to be inserted in the middle of the action, significantly disturbing the user viewing experience. In addition, in some cases the relevance of the advertisement content with respect to the video semantics is ignored.

The DEEP-AD methodology proposed in this paper notably aims at overcoming such limitations. Our proposal connects together perceptual, audio and semantic cues in order to determine the stories boundaries at a frame-level precision. In addition, given a particularly unconstrained video, our method is able to automatically determine the objects evolving within the scene and select corresponding, relevant ads. Moreover, the ads may be inserted in the video stream in the near vicinity of the related object. The following section describes the proposed approach, and details the various modules involved.

III. PROPOSED DEEP-AD APPROACH

The DEEP-AD architecture (Fig. 1) involves the following four major parts: shot boundary detection, scene identification, thumbnail extraction and ads insertion.

A. SHOT BOUNDARY DETECTION

The shot boundary detection process extends the graph partition method introduced in [32], where each frame of the video stream is considered as a vertex (node) in a graph structure. To each edge of the graph, a weight is associated with, based on the cosine distance computed between global frame descriptors. We have considered as frame descriptors the HSV color histograms, concatenated with the low level features extracted from the last layer of the ResNet50 [33] CNN architecture. A transition between two video shots is identified based on a graph partition algorithm designed to minimize an objective function within a temporal sliding window.

Let us note that a similar deep description approach is also considered in [34] where the authors propose to use deep features in order to perform the video segmentation into shots. The technique is extended in [35] were deep features are extracted and sequential patterns are analyzed through LSTM models in order to perform video shots segmentation. Each video frame is described using low level features extracted from the FC7 fully connected layer of the MobileNet CNN architecture [36]. A shot boundary is identified if the Euclidian distance between the low level descriptors is above a pre-established threshold.

The above strategies prove to be very effective in detecting abrupt transitions or gradual transitions spreading over a reduce number of frames (usually less than 10). For long gradual transitions the method requires reduced values for the decision thresholds which lead to an increased number of false alarms caused by the camera movement.

In order to overcome such a limitation, we introduce a simple, yet effective approach that helps improving the system performance. Thus, we first use low values for the decision thresholds to allow detecting gradual transitions regardless on their duration. This process makes it possible to extract a first list of frames that correspond to potential shot transitions, which is further analyzed. Then, for each potential shot transition frame, a 2 seconds video segment centered on the current frame is considered.

A regular grid sampling strategy is applied in the first frame of the video segment in order to extract a set of evenly distributed interest points.

The grid step is defined as: \( \Gamma = \frac{W \cdot H}{n} \), where \( W \) (width) and \( H \) (height) are the frame dimensions and \( n \) is the maximum number of allowed interest points. The value of parameter \( n \) controls the balance between the processing speed and the quality of the tracking process. The interest points are applied as input to the simple and very fast multi-scale Lucas-Kanade algorithm (LKA) tracker [37].

Then, we analyze the remaining number of points on the last frame of the video segment. For abrupt transitions no point will remain because the tracking cannot be performed (Fig. 2a). However, in the case of the gradual transition due to the slow variation of the video content, even if the video content changes completely between the first and the last frame of the video segment we observed that more than 50% of the total number of points remains (Fig. 2b).

In the case of large camera/object motion the number of points will decreases to less than half and tend to concentrate on a specific part of the video frame depending on the motion direction (Fig. 2c). This analysis makes thus possible to distinguish between gradual transitions and shots exhibiting high object motion.

Using the decomposition of the video stream into shots, our next objective is to detect video scenes. We formalize this as a temporally-constrained clustering problem, described in the following section.
B. SCENE/STORY SEGMENTATION

The video segmentation into scenes is based on a multi-modal, agglomerative clustering algorithm that jointly exploits visual, audio and semantic (object recognition and places identification) features (Fig. 3).

1) SHOT CLUSTERING BASED ON VISUAL APPEARANCE

a: SHOT CLUSTERING USING LOW LEVEL FEATURE DESCRIPTORS

The video shot is characterized by a uniform variation of the visual content and therefore we decided to use key-frames in order to describe the shot visual appearance.

In the same time, the use of a singular frame could result in a poor representation of the video shot due to the video content evolution in time. For this reason, we propose to extract a variable number of key-frames for each shot, depending on the temporal length of the video segment. The first key-frame associated to a shot is considered as the first frame located at a distance of 1 second after a determined shot transition. We adopted this strategy in order to avoid selecting as key-frame an image that may belong to a gradual transition (which rarely last more than 1 second). Then, we uniformly sample the shot and select for each second of video content a key-frame. More sophisticated frame sampling techniques have also been evaluated, including the selection of the most salient/discriminative frames based on the analysis of corresponding color histograms [38]. However, from our experiments, no significant improvement with respect to the uniform sampling strategy was observed.

Finally, each key-frame $i$ is represented by a global visual descriptor, denoted by $f_i$. As in the case of the shot boundary technique described in Section III.A, the descriptor $f_i$ is defined as the direct concatenation of the HSV and CNN low-level features (i.e., ResNet50). The size of the histogram descriptor is 260, while the size of the low-level descriptor extracted from the CNN layer is $7 \times 7 \times 2048$. 
In order to determine the similarity between two video shots, we need a distance between the feature vectors which can reflect with high accuracy the visual similarity.

Instead of traditional metrics (e.g., cosines distance or chi-square distance), we have considered here a different approach. More precisely, we have trained a CNN architecture and learned an embedding function \( \delta(f) \) that maps the global feature vectors into a compact Euclidean space, where distances directly correspond to a measure of visual similarity. The output of this CNN network is a 4096-dimensional global feature vectors into a compact Euclidean space, where the probability of features \( f \) being similar and takes the visual similarity score is above a pre-defined threshold, able to distinguish between different shots that share the same elements but should not be included in the same scene. However, such an approach will not guarantee that all shots assigned to the same cluster are visually similar. To this purpose, we have considered an agglomerative clustering technique, described here below.

To this end, we trained the Deep Ranking CNN [39]. A triplet based network architecture is used for the ranking loss function. The network receives as input image triplets that are fed independently into three identical deep neural networks with shared architecture and parameters. The network computes the mapping function \( \delta \).

A ranking layer is put on top in order to evaluate the loss of a triplet. The loss of the network is defined over triplets of features \( (f_i, f_j^+, f_k^-) \) and is defined by the Hinge loss as:

\[
\text{Loss} = \max(0, \| \delta(f_i) - \delta(f_j^+) \|^2 - \| \delta(f_i) - \delta(f_k^-) \|^2 + 1 - \|\delta(f_j^-)\|^2),
\]

where \( \| \cdot \| \) represents the \( L^2 \) norm in the embedding space, \( f_i \) is the current frame’s feature, \( f_j^+ \) denotes the key-frame features belonging to different shots that are assigned to the same scene, while \( f_k^- \) key-frame features belonging to shots assigned to different scenes.

To this end, we trained the Deep Ranking CNN [39]. A triplet based network architecture is used for the ranking loss function. The network receives as input image triplets that are fed independently into three identical deep neural networks with shared architecture and parameters. The network computes the mapping function \( \delta \).

A ranking layer is put on top in order to evaluate the loss of a triplet. The loss of the network is defined over triplets of features \( (f_i, f_j^+, f_k^-) \) and is defined by the Hinge loss as:

\[
\text{Loss} = \max(0, \| \delta(f_i) - \delta(f_j^+) \|^2 - \| \delta(f_i) - \delta(f_k^-) \|^2 + 1 - \|\delta(f_j^-)\|^2),
\]

The learning process has been performed using the stochastic gradient descend on mini-batches of features. For each triplet of features, we compute the gradients over the components and perform the back propagation.

Let us note that the learning process requires the availability of a learning dataset. In our case, we have considered a manually segmented dataset (cf. Section IV).

In this way, in the testing phase, a similarity measure \( p_{i,j} \) is determined for each pair of features \( f_i \) and \( f_j \) extracted from two different key-frames. The measure \( p_{i,j} \) can be interpreted as the probability of features \( f_i \) and \( f_j \) to be similar and takes values within the \([0, 1]\) interval. Finally, the visual similarity between two video shots \( s_m \) and \( s_n \) is computed as described in Eq. 3:

\[
\text{Sim}_{\text{Visual}}(s_m, s_n) = \max_{i=1,...,K(s_m)} \max_{j=1,...,K(s_n)} (p_{i,j}),
\]

where \( KF(s) \) denotes the number of key-frames extracted from shot \( s \).

Let us note that, in the context of egocentric video temporal segmentation [23], the objective is to group together video shots performed in the same location (e.g., car, office, kitchen...) at different moments in time. For commercial videos, a powerful story segmentation system needs to be able to differentiate between video shots performed in the same location by different characters.

We have adopted a CNN-based shot similarity measure in order to obtain a finegrained key-frame similarity score, able to distinguish between different shots that share the same elements but should not be included in the same scene. As typical examples that are recurrently occurring in videos, let us mention the cases of video shots performed in the same location by different characters or the one of close up shots focused on objects of the same various type (e.g., vehicles, buildings, furniture in in-door scenes). Compared with traditional metrics designed to compute the image similarity (e.g., cosines distance or chi-square distance) the proposed approach offers a gain in accuracy with more than 10%.

A first video scene segmentation is obtained by clustering together the individual shots based on their visual similarity. To this purpose, we have considered an agglomerative clustering technique, described here below.

We first impose a natural constraint: the video shots within one scene have to be temporally continuous. Then, we compute the shots similarity (cf. Eq. 3) solely within a temporal sliding window of size \( w_{size} \), defined as:

\[
w_{size} = \min \left( \sum_{i=1}^{N_{shots}} T(s_i), 60 \right) \text{sec},
\]

where \( N_{shots} \) is the number of shots considered for analysis, while \( T(s_i) \) is the temporal length of shot \( s_i \), expressed in seconds. Two video shots are assigned to the same cluster if the visual similarity score is above a pre-defined threshold, denoted by \( Th_1 \). In our experiments, the value of parameter \( N_{shots} \) is set to 5, while \( Th_1 \) is fixed to 0.9. We decided to use a high value for the \( Th_1 \) parameter in order to be certain that all shots assigned to the same cluster are visually similar. However, such an approach will not guarantee that all shots are assigned to a cluster (scene). In order to deal with the remaining shots and assign them to relevant scenes, we introduce an additional analysis, based on global face descriptors.

b: SHOT CLUSTERING USING GLOBAL FACE DESCRIPTORS

The shot clustering based on facial features involves three independent stages, including face detection, multiple face tracking and re-identification.

The face detection, performed for each individual shot, is based on the Faster R-CNN approach [40] utilized here with Region Proposal Networks (RPN) [41]. The network is initialized using a model trained on the ImageNet database [42] that is further retrained on the WIDER database [43].

The face tracking between successive frames of the video shot is performed with the help of the ATLAS algorithm introduced in our previous work [44], extended to work on multiple face instances. For each face, a low-level feature
representation is derived. The face descriptor is defined as the activation map of the last layer (before the classification stage) of the CNN architecture [45] (i.e., VGG16). The output face descriptor, of size 4096, is further normalized to a unit vector. Then, our objective is to determine with high confidence the probability of a face instance to be similar with other face instances from different video shots.

In order to perform the persons face re-identification, we propose to derive a global feature representation for each face track that is able to aggregate all face instances into a compact descriptor. In our case, the global feature $G_{\text{Face}}$ associated to a face is described in the following equation:

$$G_{\text{Face}} = \sum_i w_i \cdot f_i,$$

where $f_i$ are the normalized features corresponding to a face instance and $w_i$ is the set of real-valued, positive and unitary-normalized weight associated to the $i^{th}$ face instance.

The set of weights is determined as the output of a CNN architecture (i.e., VGG16) binary trained with only two classes that are denoted by significant and trivial [46]. The significant class includes relevant -- frontal, un-blurred and un-occluded -- face instances. The trivial class contains noisy/blurred/profile face instances whose impact on the global face descriptor needs to be minimized.

The face similarity between two video shots $s_m$ and $s_n$ is defined as:

$$Sim_{\text{Face}} (s_n, s_m) = \max_{j=1,...,A(s_n)} \left( \text{dist} (G_{\text{Face}}, G_{\text{Face}}) \right),$$

where $F(s)$ denotes the total number of faces detected in the video shots $s$, while $\text{dist}(\cdot)$ is the cosine distance between global face descriptors.

Finally, the shots are grouped into scenes based on the same agglomerative clustering technique presented in Section III.B.1.1.

At this point, we have two different predictions for the scenes boundaries: one based on low level visual features and a second relaying on global face descriptors. Finally, the clusters that share at least one common shot are merged into a single cluster.

2) SHOT GROUPING BASED ON AUDIO PATTERNS

Fig. 4 presents the synoptic scheme of the audio patterns re-identification method proposed. The module takes as input the previously determined video shots (cf. Section III.A).

The analysis is further performed on the audio stream in order to cluster together shots with similar audio patterns.

First, we split the audio stream into smaller audio segments, so-called audio chunks, using the shots timestamps. Each audio chunk is then analyzed in order to split it into homogeneous intervals, corresponding to: speech (with further differentiation into female/male voice), music, noise and silence [47].

The speaker re-identification task between various shots is treated as a multi-category classification problem. Within this context, for each audio chunk, except the silence zones, we compute spectrograms. The spectrograms are represented as vectors of size $(257 \times T \times 1)$, where $T$ is the temporal length of the audio sub-segment (expressed in milliseconds), $257$ represents the number of STFT (Short Time Fourier Transform) coefficients and $I$ is the number of audio channels used. Then, on each bin of the frequency spectrum we perform mean and variance normalization [48]. In order to perform the audio pattern re-identification in various shots we have used a modified version of the residual-network (i.e., ResNet50) architecture, specifically designed for spectrogram data, and built with an additional batch normalization stage before computing ReLUs [48].

The CNN architecture is able to ingest arbitrary time-lengths audio signals and returns as output feature vectors of fixed-length. In addition, a NetVlad layer has been added on top designed for feature aggregation and dimensionality reduction. Using the audio feature vectors $a_i$ extracted after the NetVlad layer, the audio similarity between two video shots $s_m$ and $s_n$ is computed as:

$$Sim_{\text{Audio}} (s_n, s_m) = \max_{j=1,...,A(s_m)} \left( \text{dist} (a_i, a_j) \right),$$

where $A(s)$ represents the total number of audio sub-segments extracted from the video shots and $\text{dist}(\cdot)$ is the cosine distance between descriptors.

Finally, video shots are grouped into scenes based on the agglomerative clustering technique presented in Section III.B.1.1.

So far, we have two different proposals of scenes boundaries, one given by the visual module and the second one provided by the audio one. The final scenes boundaries are obtained as the direct union between various clusters. That means that audio and video clusters containing at least one common shot are merged into a single scene.
More precisely, several situations can occur. A first case is the one where the set of shots of an audio scene is completely included in the one of a visual scene. In this case, the video scene is validated as is and the considered audio scene has no impact on the final segmentation.

In a second case, the set of shots from an audio scene overlap with the shots of two consecutive video scenes, denoted by $S_n = \{s_{n-1}, s_{n}, s_{n+1}\}$ composed of $l_n$ video shots and $S_{n+1} = \{s_{k+1}, \ldots, s_{k+l_{n+1}}\}$, with $l_{n+1}$ shots. In this case, the scene $S_n$ is extended with all the shots of $S_{n+1}$ and is defined as: $S_n = \{s_{k-l_{n+1}}, \ldots, s_k, \ldots, s_{k+l_{n+1}}\}$. The scene $S_{n+1}$ becomes void and is eliminated. Let us note that, theoretically, an audio cluster can also overlap more than two visual scenes. The above-described procedure can also tackle this situation, by eliminating all the intermediate visual scenes.

3) SHOT CLUSTERING BASED ON FOREGROUND OBJECTS

In order to deal with singular shots, not yet assigned to any scene, we propose to convey an object-based semantic representation for each shot. To this purpose, an object detection and recognition algorithm is applied on every key-frame of a video shot. We have adopted the CNN model [49] (i.e., YOLOv3) with DarkNet framework and trained on the COCO dataset [50]. The network can predict 80 classes such as: vehicles, animals, furniture, plants and other indoor/outdoor objects which can be favorably used for video story segmentation. The video shot is finally described as a union of the detected objects: $s = \{obj_1, obj_2, \ldots, obj_k\}$. Similarly, a video scene is semantically represented by the union of objects existent in its associated shots.

Singualr shots are clustered to temporally adjacent scenes if they share the same classes of objects (from the 80 categories considered) as the scene. Otherwise, a novel scene is constructed containing solely the current video shot.

4) SCENE SPLITTING USING INDOOR/OUTDOOR PLACE RECOGNITION

The perceptual and semantic features employed above are very powerful for storyline segmentation in well-structured videos, where the adjacent scenes significantly differ in terms of visual content, actors and related objects of interest. However, such techniques cannot distinguish between adjacent video scenes exhibiting the same actors/characters, but evolving in different locations. As a result, different scenes are grouped into a single one. This case notably appears in most movies (action, adventure, comedy, drama, horror, thriller, western, romance...), regardless on the movie genre.

In order to deal with such situations, various methods [51] propose to analyze the amount of speech or silence in the audio soundtrack, in order to extract the scene boundaries. Still, in complex videos such an approach usually fails, especially for continuous actions developed in various indoor/outdoor places.

In our work, we propose a different approach that emphasizes the role of the location where the scene is taking place, which defines in a certain manner the context of the given action. To this purpose, we have adopted the CNN model [33] (i.e., ResNet152) and fine-tuned it on Places365-Standard [52] dataset.

In this case, the network is focused on the background information (places, building, environment...) and can predict 365 locations (e.g., food court, cafeteria, beach, church, coast, elevator, lobby ...). The corresponding recognition probabilities are also returned.

In addition, in order to increase the robustness we have performed an initial classification of the video shots into six independent categories defined based on the video camera filming type: Extreme Wide Shot (EWS), Long Shot (LS), Medium Shot (MS), Medium Close Up Shot (MCUS), Close Up Shot (CUS), Extreme Close Up Shot (ECUS). In order to determine the shot type we have trained a ResNet50 CNN on a dataset that contains around 1000 images per each type of filming category. This categorization makes it possible to give higher importance to the indoor/outdoor location prediction performed on wide, long and medium shots, where the contextual information existent in the associated key-frames is more important (Fig. 5), since it is not occluded by foreground objects of big size.

Next, for each video shot existent in the current scene we determine the associated places. For each video shot, we retain as relevant the top $L$ locations returned by the ResNet152 CNN model. In our work, we have considered a relatively limited number of locations: $L = 5$.

Finally, we assign a label indoor/outdoor to each shot based on the majority of predicted locations.

Using the global semantic description associated to each shot of a video scene, we evaluate the possibility of splitting the scene into two independent ones. The process starts by computing the contextual similarity between adjacent, successive shots $(s_n, s_{n+1})$. The contextual similarity, denoted by $\text{Sim}_{\text{context}}(s_n, s_{n+1})$, takes into account the number of
common places between the two video shots considered and their recognition probabilities, and is defined as described in the following equation:

$$\text{Sim}_{\text{context}}(s_n, s_{n+1}) = \frac{1}{K} \sum_{i=1}^{K} \alpha(s_n) \cdot p_i^n + \beta(s_{n+1}) \cdot p_i^{n+1},$$

(8)

where $p_i^n$ (resp. $p_i^{n+1}$) denote the $i$th place recognition probability in shot $s_n$ (resp. $s_{n+1}$), $K$ is the total number of common locations recognized in shots $s_n$ and $s_{n+1}$, while parameters $\alpha$ and $\beta$ control the influence of the video camera filming type. For wide, long and medium shots the values for $\alpha$ and $\beta$ are fixed to 1, while for all the others shots the value for $\alpha$ and $\beta$ is 0.5.

A scene boundary is decided if the similarity scores between two adjacent shots is inferior to a pre-defined threshold $Th_2$. In our experiments the value for the $Th_2$ parameter was set to 0.1.

C. THUMBNAIL SELECTION FROM VIDEO SCENES

The challenge here is to characterize an entire video scene by a single thumbnail image, which corresponds to a frame that can be representative for the whole content of the scene. In order to achieve this goal, two different criteria need to be taken into account.

First, the thumbnail image has to present a high visual quality: noisy, blurry or frames affected by compression artifacts should be discarded. Secondly, in order to ensure a good visual representativeness, we need to make sure that the selected frame includes a maximal amount of information.

Concerning the visual quality, several low level features offer useful measures, including sharpness, color, blurriness information, amount of edges, motion or compression artifacts.

At the level of high level features, we can consider different criteria, such as the presence of a human face, the face size and pose, the presence of visible and easily recognizable objects or indoor/outdoor places.

In order to determine the relevance of low level features, we have designed a learning-based optimization scheme, able to assign a score to each individual key-frame depending on its visual content. We have adopted the VGG16 CNN network architecture [45], for which we have considered only two classes, denoted as relevant and irrelevant. They respectively correspond to high-quality video frames appropriate to be selected as thumbnails and low-quality ones (e.g., blurred images, video frames with various motion and compression artifacts, images with no edges...), whose impact on the thumbnail selection process should be reduced.

The CNN training is performed on a dataset with around 30000 images for each category. In order to determine the image degree of blur, we have adopted a non-referential sharpness (NRS) metric [53] that determines the local contrast in the neighborhood of the image edges, detected using the Sobel operator. Only images with a NRS value superior to 2 have been included in the relevant class. In addition, we included in the relevant class images at a resolution superior to 256 $\times$ 256 pixels and images representing aligned human faces, with little variation for the yaw, roll or pitch angles (less than 25 degrees).

In order to create the irrelevant learning class, we have synthetized a set of images by applying the following transforms: scale variation, linear motion/optical blur, scale variation and video compression noise. For the scale variation, we have considered various down-sampling factors ranging between 1/12 and 1/2 of the original image size. To model the linear motion and optical blur we have used, as suggested in [54], a kernel length that is randomly selected within the [5], [15] interval and a kernel angle ranging between 10 and 30 degrees. For generating compression artifacts we have employed JPEG compression with quality parameters randomly selected within the [10], [50] interval.

In order to determine the scene thumbnails, we apply as input to the network all the scene key-frames in order to rank the images according to their visual representativeness. The top 10 predictions have been selected for further processing.

Next, the set of candidate thumbnails are applied as input to the CNN trained based on the video camera filming type (cf. Section III.B.4). The images corresponding to wide or long shots are ranked first. Finally, we detect the semantic content of each key-frame with the help of the object detection approach introduced in Section III.B.3. The image with the highest number of object categories and characters faces is selected as the scene thumbnail.

D. ADVERTISEMENT INSERTION BASED ON SEMANTIC CRITERIAS

The key principle of the proposed advertising insertion methodology is mainly centered on the temporal segmentation algorithm presented in Section III.B, since the scene boundaries represent good candidates for the ads insertion points. However, for ensuring the optimality of the ads insertion locations, the following factors need to be carefully taken into consideration:

1. Temporal distribution—designed to estimate the ads dispersion over the video stream. The selected ads insertion points should be ideally as uniformly as possible distributed along the video timeline. In our case, since we propose to insert the ads at the level of the scene boundaries, this condition is naturally fulfilled, due to the intrinsic video structure.

In order to prioritize between various locations, we normalize the scene temporal duration with respect to the longest scene in the video. To each pair of adjacent video scenes $(S_t, S_{t+1})$ we associate a temporal relevance ($TR$) measure defined as:

$$TR_{t,t+1} = \frac{1}{2} (\bar{t}_t + \bar{t}_{t+1}),$$

(9)

where $\bar{t}_t$ and $\bar{t}_{t+1}$ represent the normalized temporal durations of scenes $S_t$ and $S_{t+1}$, respectively.

With respect to this measure, the optimal location is the one that maximizes the $TR$ parameter. In this way, we privilege...
the ads insertion between temporally longer scenes, which minimizes the risk of breaking the continuity of the action.

(2) Commercial relevance with respect to content—We propose to leverage on the video scene description in terms of associated objects and indoor/outdoor places, in order to propose ads that are semantically related to the video content. For each ad in a given commercial dataset, we apply the object/place recognition methods introduced in Sections III.B.3 and III.B.4, in order to determine its associated semantic description. Then, between each pair of adjacent video scenes \((S_i, S_{i+1})\), the ad commercial relevance \((CR_{i,j})\) is defined as:

\[
CR_{i,j+1} = \max \tilde{\theta} (S_i, ad_k),
\]

where \(\tilde{\theta}(\cdot)\) is a function designed to determine the number of common objects/places existent between the ad \(ad_k\) and the scene \(S_i\) normalized to the maximum number of common objects computed between any ad and any video scene in the movie being analyzed.

Let us note that here, solely the objects/places of the scene \(S_i\) are taken into account and not those related to the following scene \(S_{i+1}\). This strategy follows a simple causality principle and privileges ads in relationship with the content that has been already viewed.

(3) Ads intrusiveness—in order to minimize the level of ads intrusiveness, we propose to insert the ads at scenes boundary where no audio signal is present. To this purpose, we determine for each ad insertion location \((S_i, S_{i+1})\) the corresponding silence duration \((A_{i,j})\). The optimal position will maximize the silence length. The silence intervals are normalized to the maximum one.

By considering all the three factors defined here-above, we can formulate a global optimization cost associated to each pair of two successive video scenes \((S_i, S_{i+1})\) as:

\[
C(S_i, S_{i+1}) = w_1 \cdot TR_{i,j+1} + w_2 \cdot CR_{i,j+1} + w_3 \cdot A_{i,j+1},
\]

where \(w_1\), \(w_2\) and \(w_3\) are predefined weights parameters. In our experiments, we set \(w_1 = w_2 = w_3 = 1/3\), indicating that the three \(TR_{i,j}, CR_{i,j}\) and \(A_{i,j}\) factors are equally important.

The whole set of scene transitions is thus ranked by decreasing value of the cost measure defined in Eq. 11. The highest values correspond to the most optimal ads insertion locations. For each movie, the top \(k\) (with \(k\) usually varying between 3 and 5) insertion points are proposed to the video editors, which can take the final decision based on their expert knowledge.

IV. EXPERIMENTAL EVALUATION

In order to validate the performance of the DEEP-AD methodology, we have set up a detailed evaluation protocol, described in this section. We have notably focused our attention on studying the impact of the various modules involved on the overall system performance.

For the evaluation, we have considered a high variety of challenging videos.
the object detection speed and accuracy. The main idea is to boost the detection speed by reducing the number of regions of object proposal.

**a: COMPARED MODELS**

We have considered, as candidates, various deep CNN models by taking into account both the accuracy and efficiency. The selected CNN models involve: VGG16 [45], Inception v4 [55], ResNet50 [33], ResNet152 [33], Xception [56] and YOLO [49]. All tested architectures have been implemented in the TensorFlow library on a machine with one 1080Ti GPU and 64 GB of RAM.

**b: DATASET**

Using the video dataset presented in Section IV.A we have created pairs of shots that should / should not belong to the same video scene. The benchmark involves 20000 shots pairs.

**c: EVALUATION**

In the evaluation stage we have determined the probability of two video shots to be assigned to the same scene by using: (1) low level features extracted from the last layer before classification of the CNNs, (2) global face descriptors, (3) audio patterns described using spectrograms, (4) foreground objects detection and (5) place recognition. The experimental results are summarized in Table 1. At each stage, the network returning the best performances has been selected and included in the final DEEP-AD framework. However, let us note that, with some minor exceptions, all CNNs topologies return quasi similar results in terms of performances. The key issue here is to combine the various analyzers in a resilient manner, the choice of a network architecture having less impact on the overall results. In our work, we have preferred to use relatively standard CNN architectures and rather focused our attention on the adaptation strategies of each module involved.

**2) CNN TRAINING**

Based on the evaluation performed in Section IV.B.1 we present next the retained CNN architectures involved in the DEEP-AD framework together with the training processes.

For shot clustering based on visual appearance the low level descriptors are extracted from the last layer before classification of the ResNet50 architecture. The training of the CNN has been performed on the ImageNet dataset [42].

The person re-identification process is based on Faster R-CNN architecture that has been trained on a dataset with 110 celebrities taken for VGG Face dataset [46]. For each person, a maximum number of 800 face instances have been retained. We have performed the training with 50k iterations, at a learning rate of 0.0001 and a batch size of 64.

The CNN architecture necessary for the weight adaptation module uses the same parameters as the CNN involved in the face re-identification process.

The CNN architecture used for speaker re-identification contains the 110 classes as for person re-identification of which approximate half are male and half are female.

The database contains approximate 13000 speech segments belonging to the 110 actors involved. The lowest length of the speech segment admitted as input is set to 1.25 seconds. In addition, the audio segments contain, multi-speaker acoustic environments, with real world noise such as background chatter, laughter or overlapping speech.

The foreground object detection module is based on YOLOv3 CNN architecture [49] that was trained on COCO dataset [50], while the indoor/outdoor places recognition architecture is based on ResNet152 CNN model [33] that have been fine-tuned it on Places365-Standard [52] dataset.

**C. OBJECTIVE EVALUATION**

In order to perform an objective evaluation of the proposed approach, we have considered traditional metrics, widely used in the state of the art, including the recall rate (R), precision rate (P) and F1 score (F1), defined as:

\[
R = \frac{D}{D + FN}, \quad P = \frac{D}{D + FP}, \quad F1 = \frac{2 \cdot P \cdot R}{P + R},
\]

where \(D\) denotes the number of correct scenes boundaries detected by the framework (true positive scene instances), \(FN\) represents the false negative events (i.e., missed detected scenes) and \(FP\) denotes the number of false positives (i.e., scene that have been identified as new scenes but are video segments that should belong to previous/next scenes).

In order to evaluate the influence of each component over the system performance, we have considered for comparison:

1. A temporal segmentation framework into video scenes based exclusively on low level visual descriptors (color histograms represented in the HSV color space and features extracted from the last layer of the ResNet50 architecture). (2) A video scene identification method based on all the proposed visual features (low level descriptors) as in the first testing scenario extended with the person re-identification strategy introduced in Section III.B.1.2.

(3) Video segmentation based on visual and audio features. The audio patterns re-identification method proposed in Section III.B.2 is here added to the framework.

**TABLE 1. Accuracy rates of various CNN architectures.**

| Task                        | CNN Architecture | VGG16 [45] | Inception v4 [55] | ResNet50 [33] | ResNet152 [33] | Xception [56] | YOLO v3 [49] |
|-----------------------------|------------------|------------|-------------------|---------------|----------------|---------------|--------------|
| (1). Shot matching based on low level features | 0.6             | 0.67       | 0.7               | 0.7           | 0.68           | -             | -            |
| (2). Shot matching based on global face descriptors | 0.73            | 0.7        | 0.65              | 0.68          | 0.71           | -             | -            |
| (3). Shot matching based on audio patterns | 0.44            | 0.51       | 0.52              | 0.52          | 0.51           | -             | -            |
| (4). Shot matching based on foreground object detection | 0.41            | 0.48       | 0.46              | 0.48          | 0.49           | 0.49          | -            |
| (5). Shot matching based on place recognition | 0.57            | 0.63       | 0.60              | 0.66          | 0.62           | -             | -            |
TABLE 2. Experimental results of the DEEP-AD framework.

| Method                                                                 | Ground Truth Scenes | Detected (D) | False Negative (MD) | True Positive (FA) | Precision | Recall | F1 score |
|------------------------------------------------------------------------|---------------------|--------------|---------------------|-------------------|-----------|--------|----------|
| (1). Video segmentation using low level visual descriptors            |                     | 521          | 23                  | 863               | 0.37      | 0.95   | 0.54     |
| (2). Video segmentation using low level features together with the person re-identification method |                     | 490          | 54                  | 307               | 0.61      | 0.90   | 0.73     |
| (3). Video segmentation based on visual and aural features            |                     | 544          | 59                  | 202               | 0.71      | 0.89   | 0.78     |
| (4). Video segmentation based on visual and aural features together with an object recognition module |                     | 484          | 60                  | 115               | 0.81      | 0.89   | 0.84     |
| (5). Proposed DEEP-AD framework - that included visual, audio and semantic features (object and place recognition) |                     | 519          | 25                  | 117               | 0.81      | 0.95   | 0.88     |

(4) Video segmentation based on visual and audio features that include also the object detection and recognition module proposed in Section III.B.3.

(5) Semantic video segmentation based on visual, audio and semantic features extended with the indoor/outdoor place detection module (cf. Section III.B.4).

The experimental results obtained by the DEEP-AD framework are presented in Table 2. The analysis of the results highlights the following conclusions:

(1) The video shot clustering based on low level descriptors is useful when the video shots are visually similar. The method is able to identify the core of the scene. However, for precise scene boundary detection the approach usually fails.

(2) The use of global face descriptors brings a significant improvement to the accuracy of the scene identification process, with a gain of 19% in terms of F1 score. This behavior can be explained by the robustness of the extracted face descriptor that is independent to the face track length and to various types of noises that are present in the video stream.

(3) The use of the audio features proves to be effective in clustering video shots that contain the same speaking personages or the same music or noise patterns.

(4) The use of the foreground object detection and recognition framework is effective in order to group singular video shots (not assigned yet to any scene) to the adjacent scenes/stories. However, such approach independently is not optimal in the context of video scene detection. If the same type of object appears in successive scenes the method will group the two stories together, leading to a scene boundary miss-detection. For this reason, we have decided to use the shots semantic information solely to merge singular shots to their corresponding scenes.

(5) The best results (with more than 88% in terms of F1-score) are obtained by the complete DEEP-AD framework that involves audio and video analysis together with semantic interpretation of the video scenes.

By including the indoor/outdoor location information in the scene identification process, our system is able to differentiate between adjacent scenes performed in different locations by the same set of personages.

TABLE 3. Comparative experimental results of the DEEP-AD system.

| System                          | Precision | Recall | F1 score |
|---------------------------------|-----------|--------|----------|
| Sengamdu [26] - 2007            | 0.54      | 0.67   | 0.59     |
| Wang [5] - 2018                  | 0.67      | 0.72   | 0.69     |
| Kashi [20] - 2019               | 0.69      | 0.72   | 0.71     |
| Haq [22] - 2019                 | 0.82      | 0.84   | 0.83     |
| Ours                            | 0.84      | 0.95   | 0.89     |

In order to allow comparisons with the state of the art techniques introduced in [5], [20], [26] and [22] we have considered the six videos selected from the US TV series: “The Big Bang Theory”, “Friends” and “Ally McBeal”.

The experimental results obtained by the proposed framework and the systems introduced in [5], [20], [26] and [22] are presented in Table 3. We decided to compare the proposed DEEP-AD framework with the methods presented in [5] and [26] because they adopt a similar ad insertion strategy, at the level of the scene boundaries determined based on the video content analysis. In addition, in order to complete the evaluation of the temporal segmentation method, we have also compared it against two recent techniques [20] and [22] dedicated to video decomposition and based on shot transition graphs. The parameters for all method have been selected such that to maximize the performance on the testing video dataset.

As it can be observed, our approach returns gains in precision and recall of more than 6%. This behavior can be explained by the complexity and robustness of various CNN modules involved as well as by their combination within a unified workflow.

Some illustrations of the proposed temporal video segmentation framework together with the thumbnail selection and commercial insertion strategy are presented in Fig. 7. The section denoted “Shots” presents the relevant images (key-frames) extracted from a set of successive video shots.

The region “Shots Clustering” illustrates the shots grouping based on the visual, audio and semantic criteria presented above. The section denoted “Final Scenes” illustrates the video structuring generated by the proposed video temporal segmentation method.
FIGURE 7. Experimental results of the proposed video temporal segmentation method.

Shots 1 to 4 are grouped based on the visual information. Shot 5 and 9 are clustered together with all intermediary shots into Scene 2 because they contain the same set of foreground objects. Shot 10 represents a singular shot that forms on its own an independent scene (Scene 3). Shot 11 is assigned to Scene 4 based on the audio information. The shot and some parts of the scene share the same audio pattern.

The last cluster of shots is divided into two independent scenes: Scene 4 and 5 based on the place recognition methodology. For the thumbnail extraction methodology, it can be observed that all selected key-frames correspond to high-quality images. The thumbnails belong to extreme wide or long video shots and contain the maximum number of object categories relative to the entire video scene. Finally, the Ads insertion section presents the selected ads to be inserted in the video stream based on the common objects between the ad and the current video scene. The proposed ads insertion strategy is performed so that the commercials are uniformly distributed along the timeline of the video. In addition, in order to satisfy the commercial diversity the
selected ads depend on the number of common objects existent between the ad and the video scene. Scene boundaries with silent audio signal are identified and selected as optimal locations.

For the ads insertion strategy, it is difficult to establish an objective evaluation procedure, because of the highly subjective character of such process. Fig. 8 presents some examples of the results obtained.

The first scene is performed in a restaurant were the following objects have been identified: dining table, chairs and bottles. Based on this information our system decides to introduce an “Orange juice” ad, where a person is ordering some juice at a restaurant. The ad is focused on the bottle of juice.

The second scene takes place outside, where a person is driving on a field road. In this case, the system decides to introduce a car commercial. While in the final example the system is able to select the ad performed by the same actor as in the video scene (Valerie Kaprisky).

V. CONCLUSION AND PERSPECTIVES

In this paper we have introduced a novel online video advertising framework, so-called DEEP-AD, designed from the viewer’s perspective, which aims at maximizing the ads contextual relevance while minimizing the degree of intrusiveness.

The proposed approach exploits a set of computer vision algorithms and deep CNN frameworks in order to identify the optimal ads insertion points in terms of temporal distribution, degree of intrusiveness and commercial relevance. From the methodological point of view, the core of the proposed framework concerns the multimodal scene segmentation algorithm, able to detect the scene boundaries using a set of visual, audio and semantic criteria.

For each detected scene the system extracts thumbnail images, based on their degree of representativeness. Finally, the scene boundaries represent candidate ads insertion points.

The experimental evaluation performed on a video dataset with 30 elements validates the proposed methodology with
average F1-scores superior to 88%. When compared with state of the art systems [20] and [22], our approach returns gains in precision and recall of more than 6%. This behavior can be explained by the complexity and robustness of various modules involved within the proposed framework.

For further work and development we envisage extending the proposed DEEP-AD framework on a wider variety of video genres, including TV news, documentaries, talk shows or TV contests. In addition, we plan to include some sound-to-text libraries for audio signal processing in order to have a higher level of comprehension over the video/ commercial stream. Finally, we propose to test our system using regular users and to conduct a subjective evaluation protocol.

REFERENCES

[1] Comscore Releases July 2013 U.S. Online Video Rankings. Accessed: Apr. 25, 2020. [Online]. Available: https://www.comscore.com/Insights/Press-Releases/2013/6/ComScore-Releases-July-2013-US-Online-Video-Rankings

[2] W. S. Liau, K.-T. Chen, and W. H. Huo, “A illumage: Video advertising by image matching and ad scheduling optimization,” in Proc. 31st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. SIGIR, 2008, pp. 767–768.

[3] H. Liu, S. Jiang, Q. Huang, and C. Xu, “A generic virtual content insertion system based on visual attention analysis,” in Proc. 16th ACM Int. Conf. Multimedia MM, 2008, pp. 379–388.

[4] H. Zhang, X. Cao, J. K. L. Ho, and T. W. S. Chow, “Object-level video advertising: An optimization framework,” IEEE Trans. Ind. Informat., vol. 13, no. 2, pp. 520–531, Apr. 2017.

[5] G. Wang, L. Zhao, J. Li, D. Ren, and J. Zhang, “An efficient method of content-targeted online video advertising,” J. Vis. Commun. Image Represent., vol. 50, pp. 40–48, Jan. 2018.

[6] S. Yu, G. Chen, “An efficient algorithm of behavior-targeted advertising,” Comput. Appl. Softw., vol. 28, no. 1, pp. 4–7, 2011.

[7] C. Wang, P. Zhang, R. Choi, and M. D’Eredita, “Understanding consumers attitude toward advertising,” in Proc. 8th Americas Conf. Inf. Syst., pp. 1143–1148, 2002.

[8] J. Yan, N. Liu, G. Wang, W. Zhang, Y. Jiang, and Z. Chen, “How much can behavioral targeting help online advertising?” in Proc. 18th Int. Conf. World Wide Web WWW, 2009, pp. 261–270.

[9] M. Kaytoue, A. Silva, L. Cerf, W. Meiraj, C. Raissi, “Watch me playing, I am a professional: A first study on video game live streaming,” in Proc. 21st Int. Conf. World Wide Web, 2012, pp. 1181–1188.

[10] W. Hamilton, O. Garretson, A. Kerne, “Streaming on twitch: Fostering regular users and to conduct a subjective evaluation protocol.,” in Proc. 16th ACM Int. Conf. Multimedia MULTIMEDIA, pp. 1143–1148, 2002.

[11] J. Wang, B. Wang, L.-Y. Duan, Q. Tian, and H. Lu, “Interactive ads recommendation with contextual search on product topic space,” Multimedia Tools Appl., vol. 70, no. 2, pp. 799–820, May 2014.

[12] J. Wang, M. Xu, H. Lu, and I. Burnett, “ActiveAd: A novel framework of linking ad videos to online products,” Neurocomputing, vol. 185, pp. 82–92, Apr. 2016.

[13] R. Tapu, T. Zaharia, and F. Preteux, “A scale-space filtering-based shot detection algorithm,” in Proc. IEEE 26-th Conf. Electr. Electron. Eng. Isr., Nov. 2010, pp. 000919–000923.

[14] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.

[15] K. Muhammed, T. Hassanin, and S. Waik, “Efficient CNN based summarization of surveillance videos for resource-constrained devices,” Pattern Recognit. Lett., vol. 130, pp. 370–375, Feb. 2020.

[16] T. Hussain, K. Muhammad, A. Ullah, Z. Cao, S. Waik, and V. H. C. de Albuquerque, “Cloud-assisted multi-view video summarization using CNN and bidirectional LSTM,” IEEE Trans. Ind. Informat., vol. 16, no. 1, pp. 77–86, Jan. 2020.

[17] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “MobileNets: Efficient convolutional neural networks for mobile vision applications,” 2017, arXiv:1704.04861. [Online]. Available: http://arxiv.org/abs/1704.04861

[18] B. Lucas and T. Kanade, “An iterative technique of image registration and its application to stereo,” in Proc. ICAI, 1981, pp. 674–679.

[19] R. Tapu and T. Zaharia, “Salient object detection based on spatiotemporal attention models,” in Proc. IEEE Int. Conf. Consum. Electron. (ICCE), Jan. 2013, pp. 39–42.

[20] K. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, “Learning fine-grained image similarity with deep ranking,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 1386–1393.

[21] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 1–9.

[22] H. Jiang and E. Learned-Miller, “Face detection with the faster R-CNN,” in Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2016, pp. 650–657.

[23] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 248–255.

[24] R. M. Kishi, T. H. Trojan, and R. Goularte, “Correlation based feature fusion for the temporal video scene segmentation task,” Multimedia Tools Appl., vol. 78, no. 11, pp. 15623–15646, Jun. 2019.
[43] S. Yang, P. Luo, C. C. Loy, and X. Tang, “WIDER FACE: A face detection benchmark,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 5525–5533.

[44] B. Mocanu, R. Tapu, and T. Zaharia, “Single object tracking using offline trained deep regression networks,” in Proc. 7th Int. Conf. Image Process. Theory, Tools Appl. (IPTA), Nov. 2017, pp. 1–6.

[45] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Proc. ICLR, 2015, pp. 1–14.

[46] B. Mocanu, R. Tapu, and T. Zaharia, “DEEP-SEE FACE: A mobile face recognition system dedicated to visually impaired people,” IEEE Access, vol. 6, pp. 51975–51985, 2018.

[47] D. Doukh, J. Carrive, F. Vallet, A. Larcher, and S. Meignier, “An open-source speaker gender detection framework for monitoring gender equality,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Apr. 2018, pp. 5214–5218.

[48] W. Xie, A. Nagrani, J. S. Chung, and A. Zisserman, “Utterance-level aggregation for speaker recognition in the wild,” in Proc. ICASSP IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), May 2019, pp. 5791–5795.

[49] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” 2018, arXiv:1804.02767. [Online]. Available: http://arxiv.org/abs/1804.02767

[50] Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in Proc. ECCV, 2014, pp. 740–755.

[51] L. Baraldi, C. Grana, and R. Cucchiara, “Recognizing and presenting the storytelling video structure with deep multimodal networks,” IEEE Trans. Multimedia, vol. 19, no. 5, pp. 955–968, May 2017.

[52] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, “Places: A 10 million image database for scene recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 6, pp. 1452–1464, Jun. 2018.

[53] R. Ferzli and L. J. Karam, “A no-reference objective image sharpness metric based on the notion of just noticeable blur (JNB),” IEEE Trans. Image Process., vol. 18, no. 4, pp. 717–728, Apr. 2009.

[54] K. Sohn, S. Liu, G. Zhong, X. Yu, M.-H. Yang, and M. Chandraker, “Unsupervised domain adaptation for face recognition in unlabeled videos,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 5917–5925.

[55] C. Szegedy, S. Ioffe, and V. Vanhoucke, “Inception-v4, inception-resnet and the impact of residual connections on learning,” 2016, arXiv:1602.07261. [Online]. Available: https://arxiv.org/abs/1602.07261

[56] F. Chollet, “Inception: Deep learning with depthwise separable convolutions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 1800–1807.

RUXANDRA TAPU (Member, IEEE) received the B.S. degree (Hons.) in electronics, telecommunications, and information technology, and the Ph.D. degree in electronics and telecommunication from the University Politehnica of Bucharest, Romania, in 2008 and 2012, respectively, and the Ph.D. degree (Hons.) in informatics from University Paris VI-Pierre et Marie Curie Paris, France. Since 2012, she has been a Senior Researcher with the ARTEMIS Department, Institut Mines/Télécom/Telecom SudParis, France. Her major research interests include content-based video indexing and retrieval, pattern recognition, and machine learning techniques.

BOGDAN MOCANU (Member, IEEE) received the B.S. degree in electronics, telecommunications, and information technology, and the Ph.D. degree in electronics and telecommunication from the University Politehnica of Bucharest, Romania, in 2008 and 2012, respectively, and the Ph.D. degree in informatics from the University Paris VI-Pierre et Marie Curie, Paris, France. Since 2012, he has been a Researcher with the ARTEMIS Department, Institut Mines-Telecom/Telecom SudParis, France. His major research interest includes the computer application technology, such as 3D model compression and algorithm analysis in image processing.

TITUS ZAHARIA (Member, IEEE) received the Engineering degree in electronics and telecommunications and the M.Sc. degree from University POLITEHNICA Bucharest, Romania, in 1995 and 1996, respectively, and the Ph.D. degree in mathematics and computer science from University Paris V–Rene Descartes, Paris, France. He joined the ARTEMIS Department, Institut Télécom, Télécom SudParis, as an Associate Professor, in 2002, where he become a Full Professor, in 2011. His research interests include visual content representation methods, with 2D/3D compression, reconstruction, recognition, and indexing applications.