Video Instance Search for Embedded Marketing

Ting-Chu Lin*, Jau-Hong Kao†, Chin-Te Liu†, Chia-Yin Tsai‡, and Yu-Chiang Frank Wang*
* Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan
† Industrial Technology Research Institute, Hsinchiu, Taiwan
‡ Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan

***Abstract*—With the rise of online sharing platforms such as YouTube[1], advertisers become more interested in providing relevant advertisements (ads) when the embedded products are presented in videos during broadcast, so that the number of hits and potential customers will be increased. Given the product image of interest, we present a framework which allows the advertisers or video deliverers to automatically detect the embedded products throughout the video, so that relevant ads or latest product information can be delivered to the viewers accordingly. We advance the boundary preserving dense local regions (BPLR) as the local descriptors for the query and each video frame, and utilize different types of features to describe the local region. To make our framework robust yet efficient, we reduce the search space by applying the technique of inverted index, and we propose a probabilistic framework to identify the video frames in which the product of interest is presented. Experiments on TRECVID, commercial, and movie datasets confirm the effectiveness of our proposed framework.**

I. Introduction

For the Internet users and advertiser industries, online advertising remarkably gains its importance due to the rise of online media sharing platforms such as YouTube. Since the efficiency of online advertisements (ads) is determined by the number of potential customers attracted, when/how to deliver the ads while broadcasting the media content becomes a very critical issue [1]. In contrast to contextual advertising [2] which focuses on providing relevant ads in associated with the media content, embedded marketing provides the advertisers another way to place the product of interest in a context (typically in terms of video programs such as movies and TV shows) without interrupting the program.

Current online ads presented by deliverers such as YouTube are often not relevant to the video content, even if product placement occurs (see Figure 1 for example). Although the purpose of embedded marketing is not to provide relevant ads, it would be beneficial if the advertiser or the video deliverer can provide such ads/information when the embedded products are *exactly* presented. Unlike contextual advertisement, one does not need to bridge the gap between low-level contextual features and their high-level concept for recommending relevant ads. Therefore, how to properly and automatically identify the embedded products in videos so that the most relevant ads can be displayed becomes a very important task.

In this paper, we propose an instance search framework for detecting the embedded products in commercial, TV, or movie videos. Instead of extracting the semantic information of each video, we focus on the search of the embedded products so that exact time stamps can be identified when these items are presented (as illustrated in Figure 1). Given the query image of interest, which is provided by the advertisers and is embedded in the videos, we advance the boundary preserving dense local regions (BPLR) [3] to extract local regions. We consider features which describe shape, appearance, textural, and color information for each local region, and we propose a probabilistic framework to construct likelihood maps for the product of interest throughout the video. Experiments on a variety of video datasets will confirm the effectiveness and robustness of our proposed method.

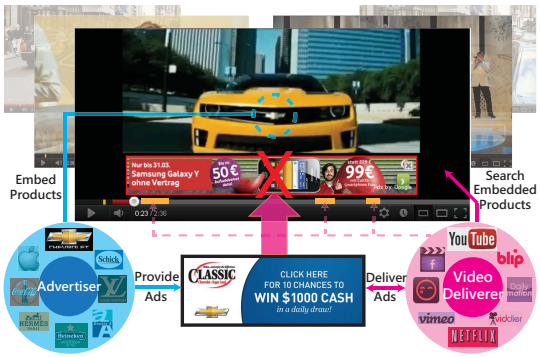

Fig. 1. Overview of providing online ads in associated with the embedded product.

II. Related Work

To search for the embedded products throughout the video based on the query image, one can divide existing approaches into three categories: *object recognition/detection*, *object retrieval*, and *image copy detection*. For object recognition or detection, one needs to collect training image data of different objects in advance for designing the classifiers or detectors [4], [5]. How to handle intra-class variations such as shape and appearance while discriminating between different object categories is the goal for the methods of this category. However, for the application of video instance search for embedded products, only one query image of the product of interest will be given, and it will be very difficult to train classifiers or detector based on a single image.

The setting of object retrieval is much closer to that of video instance search (as our work). To retrieve relevant images or video frames based on a query input, one needs to extract representative features from the query, so that the top matches from the image/video database can be identified [6], [7]. For image-based object retrieval, SIFT [8] and SURF

[1] http://www.youtube.com

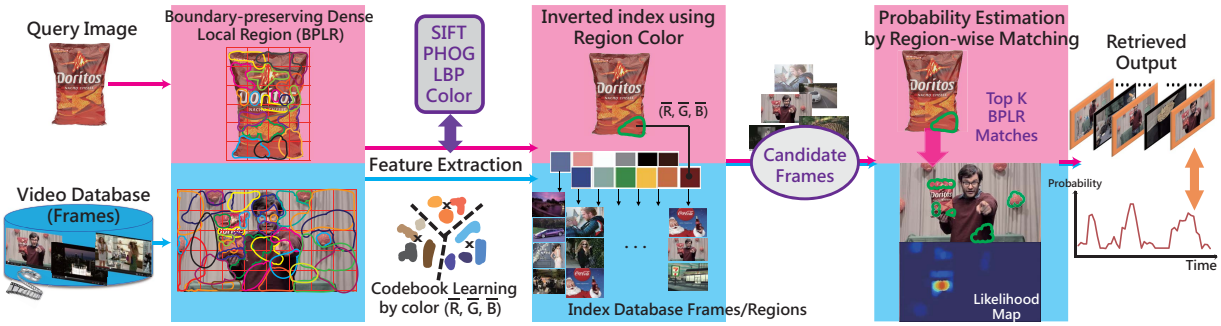

Fig. 2. Our proposed framework for searching embedded products.

[9] are among the most popular features for the task of object retrieval. For example, Philbin at el. [10] applied SIFT descriptor matching and spatial re-ranking techniques for searching similar object images. However, the use of SIFT might not be sufficient for describing objects with shape, appearance, and even video resolution variations. Therefore, how to properly describe a query input and search for the embedded items in videos is still a very challenging task.

Image copy detection aims at searching the images/videos for locating the object of interest, which is duplicated (or partially duplicated) from the query [11], [12]. Although this is closely related to video instance search for embedded marketing products, image copy detection does *not* deal with object view-point or illumination variations (only occlusion and scale changes are of concern). Therefore, it is not easy to extend this type of methods for searching embedded products in commercial videos (as we do).

III. Our Proposed Method

A. Local Region Descriptors

To describe a query image (and video frames in the database of interest), we advance segmentation based local regions and extract different types of features as local region descriptors. Although salient interest point based descriptors such as SIFT and SURF have shown their success in the task of object recognition, they are not robust to view point changes and resolution variations due to motion blur in videos. Recently, Kim and Grauman proposed Boundary Preserving Dense Local Regions (BPLR) [3], which extracts image local regions with the ability to preserve object segment/region boundaries. Each BPLR is further described by PHOG features [13], and the collection of the extracted BPLRs from the query can be used for object retrieval.

In our proposed framework, we apply dense BPLRs on both query and video frames, as shown in the first stage of Figure 2. To be more specific, for each 8×8 pixel grid in the query image and each video frame, we segment the BPLR and thus the total number of BPLRs will be dependent on the image size. Each extracted BPLR is described by its appearance (SIFT), shape (PHOG), texture (LBP [14]), and color information. The integration of these different types of features for each BPLR is to provide a robust joint feature representation, so that the proposed video instance search approach will be robust to occlusion, view-point, lighting, etc. variations (as verified later in Section 4).

B. Search Space Reduction Using Inverted Indices

Once the dense BPLRs and the associated features are extracted from the query image and video frames, we will need to detect similar BPLRs in the video frames which are similar to those in the query images. Although for the applications of embedded marketing instance search in videos, such a search process is done off-line by video deliverers, it is still preferable to reduce the search time/space for improved computational efficiency. In our framework, we apply the technique of *inverted index* for search space reduction, as illustrated in the second stage in Figure 2.

Originally proposed for text retrieval using bag-of-words (BoW) models, the technique of inverted index has recently been applied for image retrieval. Once the codebook is constructed from feature descriptors extracted from the image database, each image will be represented by a bag-of-words (BoW) model, which is a histogram representation and each attribute indicates the number of occurrences of each visual word. Once this BoW model is obtained for each image, one can consider each visual word as an entry of an inverted file, and this file records the list of images containing that visual word. As a result, given a query image (and its BoW model), one only needs to traverse the corresponding inverted files and search for the candidate images in the database. This technique can be considered as an initial filtering step and greatly reduces the search space for image retrieval.

In our work, we consider the use of color features in constructing the codebook instead of using appearance descriptors like SIFT or SURF. This is because that our instance search framework is based on BPLRs, in which the color information does not have large variations. Since the purpose of this stage is to reduce the BPLR search size for later instance search, we simply utilize color information of the extracted BPLRs in constructing the codebook and the corresponding inverted files. In particular, we calculate the average color features (RGB) for each BPLR in the video database as descriptors for codebook learning (via k-means clustering and k = 12). Once this codebook learning is complete, we use the inverted files for each visual word to traverse the video frames when performing instance search. We note that, since we do not expect the use of color information would significantly reduce the number of candidate frames, we further apply the derived inverted files to disregard irrelevant BPLRs in the filtered frames to reduce the search space. As later verified in Section

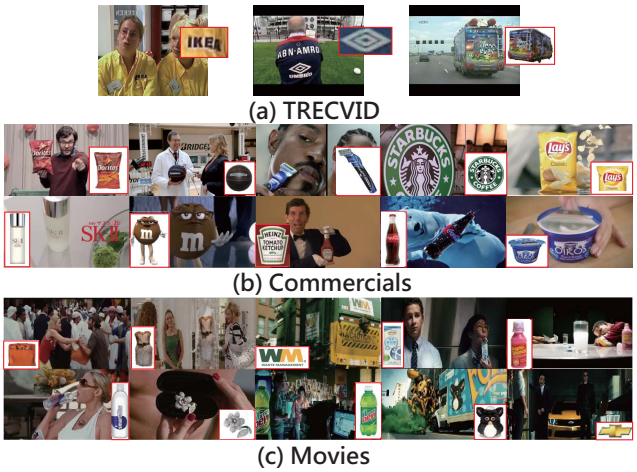

Fig. 3. Example queries and video frames.

4, a remarkably reduced number of candidate BPLRs will be resulted.

C. Probabilistic BPLR Matching for Instance Search

Once the candidate frames possibly containing the embedded products are retrieved from the database, we advance a probabilistic approach to identify the exact video frames (and the corresponding time stamps). We denote R_q as the set of BPLRs of the query q, and R_i as that for the retrieved candidate video frame i. We measure the distance between each BPLR in R_q that in R_i as

$$\text{dist}(r_q^s, r_i^t) = \|f_{r_q^s} - f_{r_i^t}\|_2, \tag{1}$$

in which r_q^s and r_i^t are the sth and tth BPLRs in R_q and R_i, respectively, while $f_{r_q^s}$ and $f_{r_i^t}$ are the corresponding joint feature representation (see Section 3.1). As a result, for each BPLR in the query image, we are able to determine the top K matched BPLR in this retrieve ith candidate video frame. Finally, we propose to derive the resulting likelihood map $L_i(x, y)$ for this frame, which is calculated as follows:

$$L_i(x,y) = \sum_{r_i^t \in R_i(x,y)} \sum_{r_q^s \in R_q} m(r_i^t, r_q^s), \tag{2}$$

$$m(r_i^t, r_q^s) = \begin{cases} \frac{1}{\text{dist}(r_i^t, r_q^s)} & \text{if dist}(r_i^t, r_q^s) \in \text{Top K shortest distances,} \\ 0 & \text{otherwise.} \end{cases}$$

In the above equation, $R_i(x, y)$ is the collection of BPLRs covering the pixel location (x, y) in the ith candidate frame. From the above derivation, we see that the proposed likelihood $L_i(x, y)$ calculates the the number of votes which the pixel location (x, y) receives from BPLR matching, while its weight inversely proportional to the distance measured between the matched pair of BPLRs. The maximum value of the derived likelihood map acts as the ranking score for that video frame in searching the relevant instances.

IV. EXPERIMENTS

A. Datasets

To evaluate our proposed framework, we consider three datasets for experiments: a subset of the TRECVID sound and vision data set, a commercial video dataset, and a movie dataset (see examples in Figure 3). The TRECVID dataset considered is a part of the instance search challenge of TREC Video Retrieval Evaluation since 2010. To adopt proper video data for the purposes of searching commercial products or trademarks, we consider three queries relevant to our task (denoted in Figure 3(a)) and the associated seven videos[2] with 325×288 pixel resolution for instance search. To conduct experiments using TV videos, we collect 10 popular commercial videos: 8 are Super Bowl commercials from 2011 to 2012, and the other 2 are collected from Asian TV programs. We search the official websites of the products[3], and use the official product images as queries. To validate the use of our method for searching embedded products in practical scenarios, we collect videos from 5 popular movies[4], which are known with embedded products presented. As for their query images, we select 10 official product images[5] whose variants are presented in the collected movie videos. We manually annotate the video frames and obtain the associated time stamps as ground truth.

TABLE I
PERFORMANCE COMPARISONS OF MAP/F-SCORES.

Dataset	SIFT Matching [17]	Spatial Coding with SIFT [12]	BPLR+ NN [3]	Ours
TRECVID	0.28/0.36	0.20/0.30	0.17/0.30	**0.31/0.38**
Commercials	0.45/0.56	0.35/0.50	0.41/0.54	**0.55/0.61**
Movies	0.26/0.40	0.23/0.37	0.29/0.43	**0.41/0.53**

We apply the code of [3] to segment the images for obtaining BPLRs as our local image descriptors. We consider SIFT, PHOG, and LBP features to describe each BPLR. For SIFT, we extract dense SIFT descriptors for each BPLR, and the collected SIFT descriptors are encoded by a codebook of 600 visual words. Thus, a 600-dimensional vector will be resulted. For PHOG, we consider 3 pyramid levels and 8 orientation bins for each BPLR, and obtain a $(1 + 2^2 + 4^2) \times 8 = 168$ dimensional vector. To describe textural information, we apply the tools developed in [15] to extract the LBP features, and result in a 58-dimensional LBP feature vector for each BPLR. As for the color feautures, we calculate the histogram for each BPLR in the CIELab space. This color space has been observed to provide promising discriminating ability as suggested by [16]. Applying the tools in [3], we quantize each channel into 23 bins and derive a 69-dimensional vector to describe the color information for each BPLR. Once all features are obtained, we normalize and concatenate each of them as the final BPLR descriptor.

B. Discussions

We compare our method with three popular approaches: SIFT matching for image retrieval [17], spatial coding with SIFT for searching partial-duplicated images [12], and BPLR matching with HOG using Nearest Neighbor (NN) [3]. For the task of video instance search for embedded marketing, since advertisers have the prior knowledge of the video of interest (i.e., those with products embedded), we only consider the

[2] Video id: BG_2403, 3288, 34858, 36021, 36866, 38164 and 38421

[3] Doritos, Bridgestone, Gillette, Starbucks, Lays, SK-II, M&M'S, Heinz, Coke and Oikos.

[4] Sex and the City ('08), Sex and the City 2 ('10), Transformers ('07), Transformers: Revenge of the Fallen ('09) and Transformers: Dark of the Moon ('11)

[5] Hermes (bag), Matthew Williamson (dress), Waste Management, YiLi Nutrition Full-Fat ShuHua Milk, Pepto-Bismol, Glaceau (Smart Water), Ramona M Boucher (ring), Mountain Dew, Furby and Chevrolet.

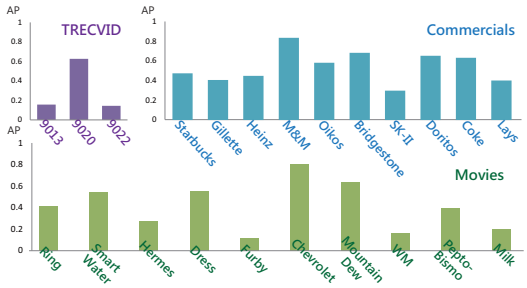

Fig. 4. AP of different queries using our method.

query and its corresponding video when performing the search. For each query, we compute the precision-recall curve using the ranking results as discussed in Section 3.3, and compute the average precision (AP) and the highest F-score. Note that precision is calculated as the ratio of the retrieved positive frames to the total number of retrieved frames, and recall is the ratio of retrieved positive frames to the total number of positives in a video. We consider the mean average precision (MAP) and the average F-score are the metrics for evaluation.

Table I lists the performance of different methods. We see that our method utilizing multiple features with a probabilistic searching algorithm based on BPLR achieved the best performance. The highest MAP and F-score were obtained for the commercial dataset. This is because that the products in commercial videos are typically meant to be explicitly presented for the viewers, and thus better retrieval results can be expected. Although the products of interest in the other two datasets exhibit more variations and make the matching task more challenging, our method still achieved the best results among all approaches considered. Figure 4 shows individual AP scores for each query using our approach.

We now discuss the effectiveness of our search space reduction technique (as discussed in Section 3.2). From our experiments, we observe that we disregarded about 9% of the video frames for a 1.5-minute and 300×250 pixel resolution video (given a 100×100 pixel query image). To further reduce the search space for BPLR matching, we applied this technique to filter the BPLRs in those candidate frames. In our experiments, only about 12.8% of the BPLRs were resulted (49M out of the original 389M), and thus the number of candidate BPLRs to be searched was significantly reduced. As for runtime estimates (in Matlab), the average computation time is 0.486 second per frame on an Intel Core 2 PC with 2.66 GHz processors and 4G RAM.

Figure 5 shows example video frames and the derived likelihood maps. Although the products in this figure are with severe scale changes, partially occluded, distorted, or even presented in a very cluttered background, our method still produced promising likelihood outputs which allowed us to identify the video frames and the associated time stamps.

V. Conclusions

In this paper, we presented an instance search framework for identifying products of interest embedded in videos. Once the query image is given by the advertiser, our method aims at automatically retrieving the video frames (and the corresponding time stamps) in which the target products are presented and possibly with scale, view-point, occlusion, etc. variations. Using dense BPLRs as local descriptors, our method extracts and describe each BPLR in terms of different types of features for better representation of queries and video frames. To alleviate the computational cost, the search space for BPLR matching is further reduced using inverted indices with color information. Finally, the proposed probabilistic framework for BPLR matching is able to identify the embedded products throughout video frames. Experimental results on a variety of video datasets verified the effectiveness of our method, and its success would allow future advertisers or video deliverers to display relevant ads or latest product information accordingly.

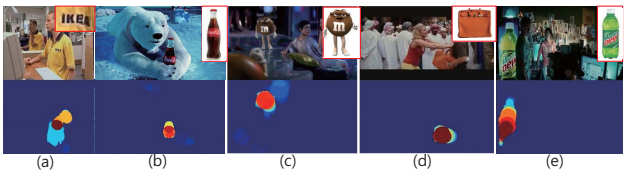

Fig. 5. Example video frames and the corresponding likelihood maps produced by our method.

Acknowledgment

This work is supported in part by Industrial Technology Research Institute, Taiwan.

References

[1] X.-S. Hua, T. Mei and A. Hanjalic. Online multimedia advertising: techniques and technologies. *IGI Global*, 2011.
[2] B. Ribeiro-Neto et al. Impedance coupling in content-targeted advertising. In *SIGIR*, 2005.
[3] J. Kim and K. Grauman. Boundary Preserving Dense Local Regions. In *IEEE CVPR*, 2011.
[4] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-Based models. *IEEE PAMI*, 2010.
[5] S. Belongie, J. Malik and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE PAMI*, 2002.
[6] J. Sivic and A. Zisserman. Video Google: a text retrieval approach to object matching in videos. In *IEEE ICCV*, 2003.
[7] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: automatic query expansion with a generative feature model for object retrieval. In *IEEE ICCV*, 2007.
[8] D. Lowe. Object recognition from local scale-invariant features. In *IEEE ICCV*, 1999.
[9] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. SURF: Speeded Up Robust Features. In *ECCV*, 2006.
[10] J. Philbin et al. Object retrieval with large vocabularies and fast spatial matching. In *IEEE CVPR*, 2007.
[11] Z. Wu et al. Bundling features for large scale partial-duplicate web image search. In *IEEE CVPR*, 2009.
[12] W. Zhou et al. Spatial coding for large scale partial-duplicate web image search. In *ACM Multimedia*, 2010.
[13] A. Bosch, A. Zisserman, and X.Munoz. Representing Shape with a Spatial Pyramid Kernel. In *ACM CIVR*, 2007.
[14] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns In *IEEE PAMI*, 2002.
[15] *http://www.vlfeat.org*.
[16] M.-M. Cheng et al. Global Contrast based Salient Region Detection. In *IEEE CVPR*, 2009.
[17] D. Lowe. Distinctive image features from scale-invariant keypoints. In *IJCV*, 2004.