

What Is Happening in The Video? —Annotate Video by Sentence

Xueming Qian, *Member, IEEE*, Xiaoxiao Liu, Xiang Ma, Dan Lu, and Chenyang Xu

Abstract—Due to the popularity of online video sharing websites such as YouTube, millions of users have treated online video as a source of information and entertainment. So Video annotation has evoked great interest in the past few years. In this paper, we propose a four-step approach to automatically annotate video shots with sentences. The first step is video preprocessing, converting video shot into a sequence of frame images. The second step is to find related candidate elements of the sentence about the video contents. The main elements in the sentence are objects, events, scenes, and modifiers. These candidate elements are gained by searching for similar images with the video frames in our collected image datasets instead of video datasets. The third step is to select the best elements among these candidate ones by a weighted scoring algorithm. The final step is to construct a sentence with the help of a correlation graph algorithm to analyze the relationships among the best elements. The experimental results indicate that our method is effective to annotate videos with sentences. What is more, the weighted scoring algorithm and the correlation graph algorithm that we propose are efficient in developing the experimental performance.

Index Terms—Algorithm, Image dataset, Sentence element, Video annotation, YouTube

I. INTRODUCTION

WITH the prevalence of social multimedia in the 21st century, digital images have become more and more accessible to the public. As time goes by, however, simple images can no longer meet peoples' demand and other more informative media are needed. So videos become increasingly popular. And technologies assisting users to search and understand contents of videos are required.

Conventional approaches to video annotation predominantly focus on supervised identification of a limited set of concepts. However, many ambiguous meanings will be introduced when

only the keywords provided for searching the video. Therefore, studies have been conducted on annotating videos with sentences. Since video content includes objects, events and scenes, generating sentences for videos will help to better understand the underlying activities happening in the video and there will be no ambiguity introduced. Moreover, if a video is annotated with a sentence, it's easier for users to search with flexible queries.

While the idea of annotating videos with sentences is promising, there are several challenges. First of all, contents of online videos are too complex to describe artificially, so let alone automatically generate a sentence. Second, it is not easy to accumulate videos as training datasets since most of the online videos have no labels. Meanwhile, it is time-consuming to manually annotate a huge amount of videos with sentences. Last but not least, representing contents of videos with natural languages is more convoluted and multifaceted compared with independent tags: it needs not only to estimate objects in videos but also objects' actions and scenes of events. What is more, the correct grammar is another factor we should consider of.

There are some pioneering works in [1] and [2] concentrating on generating sentences for videos. [1] introduces a novel two-step framework for textually annotating unconstrained videos: visual similarity video matching at first and then an annotation analysis that employs commonsense knowledge bases. After comparing the dominant low-level features from the query video with the corresponding ones of videos from a pre-annotated dataset, annotations of the most closely-matched videos are selected as the candidate ones. Then the final annotation is obtained by exploiting the semantic relationships between the terms used in the candidate annotations. In this way, it generates a simple sentence to describe the contents of a video. In the paper [2], Bardu et al. present a system producing sentential descriptions of a video. At first, humans in the video are detected and tracked. Then they recognize actions of humans in virtue of a trained human body-posture codebook. At last, via a detected action class and the associated tracks which are based on the templates built from action classes, they produce a sentence. Their generated sentences are both accurate and structurally complex: these sentences can not only delineate an objection's direction such as "from the left" and "leftward" but also apply an adverb to describe the object's velocity like "slowly" or "quickly" or an adjective modifying the object's shape, such as "tall" and "narrow", which is realized by the utilization of object detection and tracing. In the paper [23], Tan et al. try to recount videos' contents with audio-visual concept classifiers and their machine-generated descriptions are pretty informative. Their video content recounting framework

This work is supported in part by the Program 973 No. 2012CB316400, by NSFC No.60903121, 61173109, 61202180, 61332018, Microsoft Research Asia, and Fundamental Research Funds for the Central Universities (No. 310824153508).

Xueming Qian (corresponding author, qianxm@mail.xjtu.edu.cn), Xiaoxiao Liu (liuxiaoxiao266@mail.xjtu.edu.cn), Dan Lu, and Chenyang Xu are with the SMILES LAB at School of Electronics and Information Engineering, Xi'an Jiaotong University, 710049, China.

Xiang Ma is with the School of Information Engineering, Chang'an University, Xi'an 710048, China (e-mail: maxiangmail@163.com).

consists of two components. First they learn the audio-visual concepts in the video and then generate the rule-based textual descriptions. Experiments are conducted on 7,156 10-second clips from 565 training videos with each clip manually labeled.

In our work, we focus on the unconstrained video data downloaded from online video portals such as YouTube. The content of these videos are very diverse in theme and sophisticated in content, which makes our sentence generation more challenging. Besides, experiments are conducted on the unprocessed video shots and NUS-WIDE [29] image dataset.

As we all know, videos are composed of a sequence of images. In [3], Yang et al. propose the idea that in the process of video tagging, image dataset can be used as the training data by transfer learning. In the work [4], the authors try to tag tags. That is to say, annotating tags with more property tags like location, color and so on. In the process of tagging color, texture and shape, they also choose positive images as their training data to search from.

In our work, we focus on the unconstrained video data downloaded from online video portals such as YouTube. The contents of these videos are sophisticated and diverse in theme, making our sentence generation more challenging. Besides, experiments are conducted on the unprocessed video shots and NUS-WIDE [29] image dataset.

The contributions of this paper can be described as follows:

(1) We propose an automatic sentence annotation approach for free style user homemade video on the video portals such as YouTube and Flickr rather than on a fix format video such as surveillance video. We propose to gain a series of descriptive vocabularies for the video shot and generate a sentence to state the topic of the video.

(2) We use user-annotated image datasets as the training data to avoid the costly acquisition of a manually annotated video training set. As we all know, the biggest obstacle we face in video annotation is the lack of well-labeled training videos. While in our work, we use images with user-generated tags instead to avoid this problem.

(3) We put forward a weighted scoring algorithm and a correlation graph algorithm to optimize our experimental performance. In the process of sentence generation, we propose a weighted scoring algorithm to verify the accuracy of sentence elements and a correlation graph algorithm to guarantee the rationality of the sentence. These two algorithms are vital in developing the performance of our experiments.

(4) We introduce an interesting method to construct the sentence which converts the complicated, time consuming task into a simple Crossword puzzles.

The remainder of this paper is structured as follows. In Section II, we review the related work on video annotation. Our approach is illustrated in Section III. The experimental setup and performance are shown in Section IV. We make some discussions in Section V. In Section VI, the conclusions and future work are given.

II. RELATED WORK

Video annotation (also widely known as video concept detection or high-level feature extraction), which aims to automatically assign descriptive concepts to video content, has received intensive research interests over the past few years.

Various methods are put forward to automatically annotate videos with words. There are several works with respect to TRECVID [5], an annually video retrieval contest with the goal of creating a stock of best practice for video retrieval. In [6], Wang et al. propose a learning based approach for video annotation. They learn the concepts in videos by using the graph fusing following factors: multiple modalities, multiple distance functions, and temporal consistency. In [7], they propose a novel semi-supervised learning algorithm, named semi-supervised learning by kernel density estimation, which is based on a non-parametric method, and therefore the “model assumption” is avoided. In [8], Moxley et al. aim to exploit the overlap in contents of news video to automatically annotate by mining similar videos that reinforce, filter, and improve the original annotations. [9] propose a novel method named correlative linear neighborhood propagation to improve annotation performance. The amount of online videos is very huge. So some researchers focus their works on online videos. Ulges et al. proved that content-based tagging can be learned from user-tagged online videos such as videos contributed by YouTube [10]. Moxley et al. present an approach to recommend multimedia with new annotations and filter existing incorrect annotations in [11]. In [12], the authors present a system called Polemic Tweet to annotate and analyze videos through tagged tweets. The authors of [13] propose an intuitive method called Walkie Tagging for video annotation based on spoken words in the mobile environment. The work in [14] is a creative work on the application of online videos, in which Mei et al. try to model and mine users’ capture intention for homemade videos.

Some work on video annotation only concentrate on one kind of videos like sports video, traffic video or surveillance video. Li et al. propose an efficient method to annotate products in videos in [15]. It collects a set of high-quality training data by mining information from Amazon and Google to build visual signature for each product. Then noise is removed by a correlative sparsification approach to refine the visual signatures which are used to annotate video frames. [16] present a software application for annotating traffic videos with ground truth.

Most of the methods on image or video annotation only generate nouns as their results. How to decide the verb, however, is still an obstacle in sentence-making. So some previous works have focused on this aspect. In [17], Sun et al. put forward an algorithm on verb-object image classification via hierarchical nonnegative graph embedding. They divide the verb-object images into separate groups if they share the same object part while different verb part. In [18], Tian et al. propose a data-driven approach to verb oriented image annotation. At first, they obtain verb candidates by generating search queries for a given image with initial noun tags and establishing a sentence corpus from those queries. Then they further re-rank the candidate verbs with the tag context discovered from the images both semantically and visually similar to the given image in the MIR Flickr dataset.

The above works aim at annotating images or videos with words. There are also some works that aim at finding textually descriptions for images or videos [40-44]. In [40, 41], a latent-community and multi-kernel learning based approach is proposed to annotate images automatically. Community detection method is applied to cluster these concepts as

communities. Multi-kernel learning SVM is introduced to specify the communities and extract meaningful entities with some simple features. In [43], a latent structure between correlated semantic concepts are exploited in annotation models by using both context and content information. In [19], Ushiku et al. try to understand images with natural sentences. They examine captions of images similar to an input image, and generate a sentential description to the input image by mining the relationships between the texts and reconstructing the captions. Farhadi et al. build a system to compute a score linking an image to a given image in [20]. The sentence with the highest score is recommended to describe the contents of the given image. In [21], Li et al. propose the first attempt to classify events in static images by integrating scene and object categorizations. They classify the event in sport games as well as to provide a number of semantic labels to the objects and scene environment within the image. In [22], Yao et al. present an image parsing to text description framework to generate text descriptions of image and video content based on image understanding. Firstly, they use semi-automatic method to parse images from the Internet in order to build an and-or graph for visual knowledge representation. Secondly, they use automatic methods to parse image/video in specific domains and generate text reports useful for real-world applications. Their study about videos focuses on maritime and urban scene video surveillance and driving scene understanding. Tan et al. describe the complex video contents textually by using audio-visual concept classifiers [23]. They use the audio-visual classifiers to determine the video concept in their concept library and generate descriptions to recount the video content with a set of templates. They have achieved promising results. Compared with our work, however, their concept library is relatively small. So, videos recounted by their framework are very limited. For example, only six human action concepts are used in their experiments: walking, running, squatting, standing up, making stuff with hands, and batting baseball.

Videos are explained with sentences in [1]. Their work is reasonable and they have achieved great experiment results. However, their training video datasets are pre-annotated with their manual generated sentences. Surely, it is a time-consuming work. While in our method, there is no need to manually annotate the videos, because what we use is an image dataset along with user-contributed tags instead. What is more, they search for similar videos with the query video in order to get candidate annotations. However, we try to find similar images with frames in the video shot like content based image retrieval (CBIR).

In the paper [2], Bardu et al. present an algorithm that makes video in and sentence out. However, their test videos are almost all surveillance videos about human's activity under basically the same background, which makes object detection and tracking relatively easy to achieve. Compared with their work, our video dataset are more diverse in themes and sophisticated in contents. The video dataset is not only about human action. We recognize the object, event, scene and adjective of the video by searching for visually similar images instead of object detection and tracking. Content-based image retrieval [24] is the foundation of our method. Content-based image retrieval provides a lot of useful techniques and is strongly related to video annotation via the use of key frames. Also, we testify the

accuracy and correlation of these sentential elements with two algorithms. At last, these sentential elements are combined along with article, link verb, and preposition to complete a whole sentence.

Video categorization has also drawn many attentions in recent years. Wu et al. determine the category of web video by combining three aspects: semantic meaning, video relevance and user interest [25]. In [26], Yang et al. add two modalities on the basis of low level features: semantic modality, including three feature representations, i.e., concept histogram, visual word vector model and visual word Latent Semantic Analysis (LSA), and surrounding text modality including the titles, descriptions and tags of web videos. In the paper [27], Yuan et al. have presented a novel method for automatic video genre categorization utilizing spatial-temporal low-level features. They first define a hierarchical and relatively comprehensive ontology for video genres, and then propose a novel hierarchical SVM scheme for genre categorization, in which a series of SVM classifiers are dynamically built up in a binary tree form and optimized locally or globally.

III. OUR APPROACH

A. Overview of Our Approach

The process of our approach can be divided into four steps as shown in Fig.1. The first step is video preprocessing. The second step is candidate sentence elements acquisition by similarity measurement between query video frames and images in our datasets. Images of our datasets are in four domains: object, event, scene, and modifier (adjective). Tags of the closest visual neighbors are chosen as the candidate sentence elements. The third step is selecting the elements that best describe the contents of the query video using a weighted scoring algorithm. The last step is analyzing the relationships between the selected elements by a correlation graph algorithm and constructing the sentence.

B. Video Preprocessing

To cope with the large amount of online videos, we transform every video shot into a set of images by extracting one frame every one second.

Let V denotes the video shot we downloaded from YouTube. If the video shot lasts for T seconds, then we extract a frame each second and get a cluster of images: $I = \{I_1, I_2, \dots, I_T\}$, where $I_i = V|_{time=i}$, $i = \{1, 2, \dots, T\}$.

For each image I_i , we extract their low-level visual features f_i . Finally, we get a set of low-level features for the image cluster $F = \{f_1, f_2, \dots, f_T\}$. The specific type of visual features is shown in Section IV.

C. Candidate Sentence Element Acquisition

Given a query image cluster $I^q = \{I_1^q, I_2^q, \dots, I_T^q\}$ extracted from the query video shot V , it includes T images. The features of images in this cluster are $F^q = \{f_1^q, f_2^q, \dots, f_T^q\}$. We first find the candidate sentence elements that may describe video contents. These candidate sentence elements include four domains: object (O), event (E), scene (S) and adjective (A).

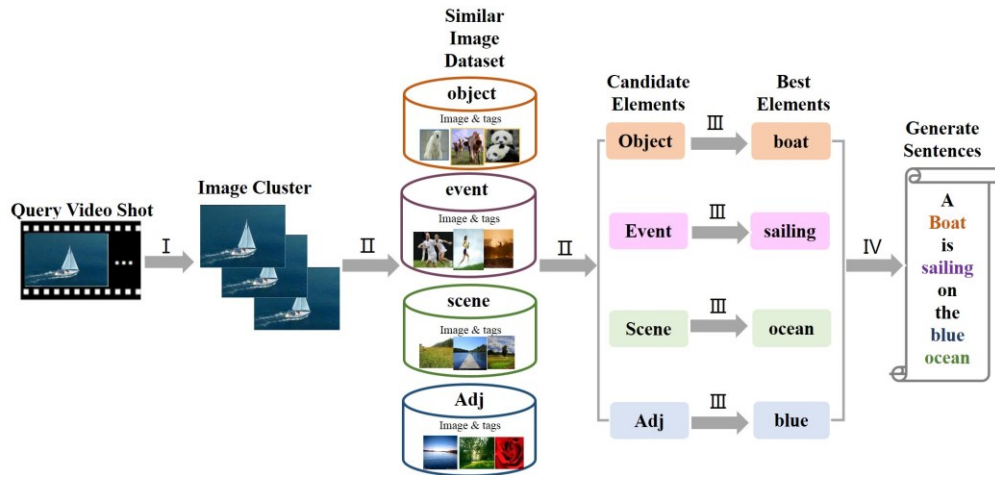


Fig. 1. Overview of our approach. I is the preprocessing of the query video shot. II is candidate sentence elements acquisition by similarity measurement between query video frames and images in our datasets. III is elements selection with a weighted scoring function. IV is sentence generation with a correlation graph algorithm.

For the element object, event, scene and adjective, we downloaded a series of images to explain its content respectively: object image dataset (OI), event image dataset (EI), scene image dataset (SI) and adjective image dataset (AI).

$$OI = \{I_i^o\}_{i=1}^M \quad (1)$$

$$EI = \{I_i^e\}_{i=1}^N \quad (2)$$

$$SI = \{I_i^s\}_{i=1}^X \quad (3)$$

$$AI = \{I_i^a\}_{i=1}^Y \quad (4)$$

where M , N , X , and Y is the number of images in OI, EI, SI, and AI respectively. Each image in the dataset OI, EI, SI, and AI has only one tag. The tag is a word describing an object, event, scene or adjective in the vocabulary table (TABLE I).

We extract the features of images in OI, EI, SI and AI. The visual features of them are described as follows:

$$F^c = \{f_i^c\}_{i=1}^{M_c}, \quad c = \{o, e, s, a\}, \quad M_o = M, M_e = N, M_s = X, M_a = Y \quad (5)$$

M_c is the number of images in the set c .

For the i -th image $I_i^q \in I^q$ of the video shot, we search for its nearest neighbors by measuring its Euclidean distance with images in OI, EI, SI and AI.

$$D_i^c(j) = \|f_i^q - f_j^c\|, \quad j = 1, 2, \dots, M_c, \quad c = \{o, e, s, a\} \quad (6)$$

where $\|*\|$ denote the Euclidean distance of vector $*$.

Then, we rank the distances $D_i^o(j)$, $D_i^e(j)$, $D_i^s(j)$ and $D_i^a(j)$ in ascending order and select the top ranked R images in the dataset as its visual neighbors. We set $R=10$, and we discuss it in Section IV. Thus for these four domains, we find their visual neighbors and denote them as VN_i^o , VN_i^e , VN_i^s , and VN_i^a .

$$VN_i^c = \{I_j^c\}_{j=1}^R, \quad c = \{o, e, s, a\} \quad (7)$$

where j means the j -th top ranked similar image in the image dataset. For example, every image in VN_i^c , it has a tag in the corresponding domain. These are the candidate sentence elements w_i^c for $I_i^q \in I^q$. So for every image in the cluster, we have elements in each of CE_i^o , CE_i^e , CE_i^s , and CE_i^a , which are denoted as follows:

$$CE_i^o = W(VN_i^o) = \{w_{i,j}^o\}_{j=1}^R \quad (8)$$

$$CE_i^e = W(VN_i^e) = \{w_{i,j}^e\}_{j=1}^R \quad (9)$$

$$CE_i^s = W(VN_i^s) = \{w_{i,j}^s\}_{j=1}^R \quad (10)$$

$$CE_i^a = W(VN_i^a) = \{w_{i,j}^a\}_{j=1}^R \quad (11)$$

where $W(*)$ means to acquire the tags of all images and $w_{i,j}^c$ denotes the tag of the j -th ranked similar image of the image i in the domain c , $c = \{o, e, s, a\}$.

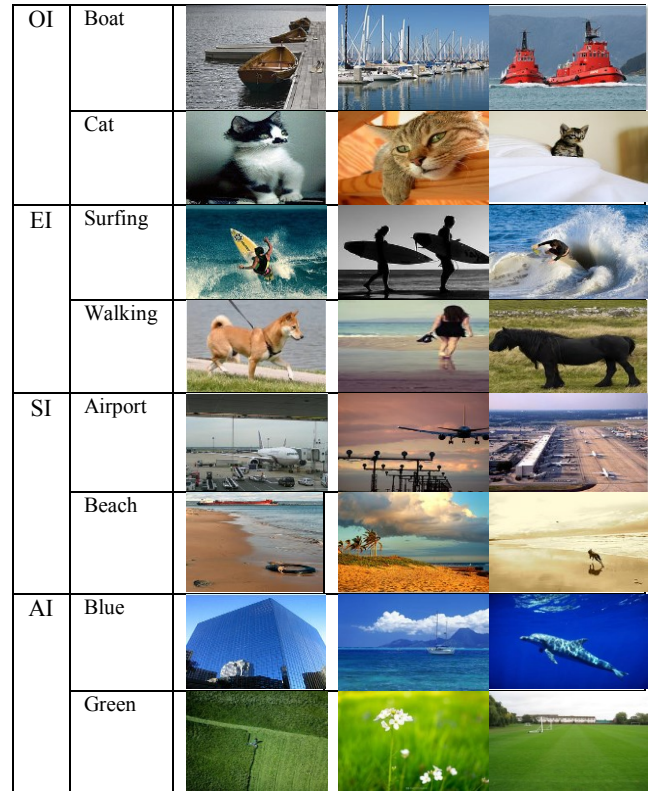


Fig. 2. Some of the examples in the four image datasets: OI, EI, SI, and AI.

So we get four types of candidate elements (object, event, scene and adjective) for every image in the cluster I^q .

In Fig. 2, we have shown some images as examples of our datasets OI, EI, SI and AI.

D. Best Sentence Element Selection with a Weighted Scoring Algorithm

Among all the candidate sentence elements, we conduct a weighted scoring algorithm to select the best element in each domain.

In term of the image cluster $I^q = \{I_1^q, I_2^q, \dots, I_T^q\}$, we have gained four sets of candidate sentence elements.

$$CE^c = \{CE_i^c\}_{i=1}^T, \mathbf{c} = \{o, e, s, a\}. \quad (12)$$

Now we try to select the tag that can best describe the cluster's content with calculating the relevance score. Let the key-frame in this video shot be I_k^q . Clearly the tag of I_k^q should be given with the highest weight. $I_{k+1}^q, I_k^q, I_{k-1}^q$ are adjacent to each other in time sequence. Taking the temporal consistency of the video contents into consideration, the weights of images near I_k^q should be given with higher weights, the image far from I_k^q should be given a lower weight. The principle of weight given is shown as follows.

$$weight(w) = \begin{cases} 1, & w \in W(I_k^q) \\ \alpha, & w \in W(I_{k+1}^q) \text{ or } w \in W(I_{k-1}^q) \\ \beta, & \text{others} \end{cases} \quad (13)$$

where parameters α and β are positive numbers and we set $\alpha=0.8$, and $\beta=0.5$. The discussions for them are illustrated on section IV.

In the four domains, the relevance score is calculated as follows.

$$score(w_i^o) = \sum_{w_i^o \in CE^o} weight(w_i^o) * C(w_i^o) \quad (14)$$

$$score(w_i^e) = \sum_{w_i^e \in CE^e} weight(w_i^e) * C(w_i^e) \quad (15)$$

$$score(w_i^s) = \sum_{w_i^s \in CE^s} weight(w_i^s) * C(w_i^s) \quad (16)$$

$$score(w_i^a) = \sum_{w_i^a \in CE^a} weight(w_i^a) * C(w_i^a) \quad (17)$$

where $C(*)$ is to count the occurrence number of one tag in the image.

We rank the scores in the descending order. The selected element is the one with the highest relevance score for the corresponding element. Thus we have:

$$w_c' = \max score(w_i^c), \mathbf{c} = \{o, e, s, a\}. \quad (18)$$

These words will probably be the main parts of our recommended sentence.

E. Sentence Generation with a Correlation Graph Algorithm

We will refine the selected elements by a correlation graph algorithm. We also take the relationships among these three elements object, event and scene into consideration.

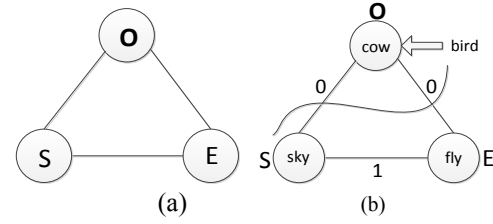


Fig. 3. The illustration of correlation graph algorithm. (a) The full connected undirected graph of these three elements. (b) The example of correlation graph algorithm.

A graph consists of a set of nodes and a set of edges that connect the nodes. We model these three elements by a full connected undirected graph with only three nodes as shown in Fig. 3(a). The edge between two nodes measures their semantic correlation modeled by the Normalized Google Distance (NGD) [28].

The NGD is a semantic similarity measure derived from the number of hits returned by the Google search engine for a given set of keywords. Keywords with the same or similar meanings in a natural language sense tend to be “close” in units of normalized Google distance, while words with dissimilar meanings tend to be farther apart.

Specifically, the normalized Google distance between two search terms x and y is

$$NGD(x, y) = \frac{\max\{\log f(x), \log f(y)\} - \log f(x, y)}{\log N - \min\{\log f(x), \log f(y)\}} \quad (19)$$

where N is the total number of web pages searched by Google; $f(x)$ and $f(y)$ is the number of Web pages containing search terms x and y , respectively; $f(x, y)$ is the number of web pages on which both x and y occur.

If the $NGD(x, y) = 0$ then x and y are viewed as alike as possible, but if, $NGD(x, y) \geq 1$ then x and y are very different. If the two search terms x and y never occur together on the same web page, but do occur separately, the NGD between is infinite. If both terms always occur together, their NGD is zero.

$$Edge = \begin{cases} 0, & NGD(x, y) \geq 1 \\ 1, & NGD(x, y) < 1 \end{cases} \quad (20)$$

The relationship between and NGD is introduced in Eq. (20). If, we set the weight of this edge to be 0, which means there is no semantic correlation between these two nodes. If, we set the weight of this edge to be 1, which means there is semantic correlation between these two nodes. If two edges of one node are all 0, then we will give up this node and choose another concept with the second high score as the new node. The example is given in Fig. 3(b). As we can see, the word “**cow**” is not related to the other two words “**sky**” and “**fly**”. So we exchange it with the word “**bird**”. As a result, we have got three final tags $w^* = \{w_o^*, w_e^*, w_s^*\}$. Along with $w_a^* = w_a'$, these four tags are used to generate a sentence.

After getting the final tags $w = \{w_o^*, w_e^*, w_s^*, w_a^*\}$, we are ready to generate a sentence with them. The proposed video sentence generation approach is as follows. The object part is at the beginning of the sentence. The key problem is to determine the articles like “**a**”, “**an**”, or “**some**”. While in the event part, we use “**be doing**” to compose the main structure of the

sentence. We need to decide the type of link verb such as is and are. At last, in the scene part, we need to tell the relationship between the object and the scene in order to use the correct preposition like “in”, “on” and so on. What is more, we have an adjective to modify scene. So, we are required to find the article, link verb, and the preposition. These three parts are denoted as $\{a, l, p\}$. These problems are tackled by using basic knowledge. For example, word with an “s” or “ies” in the end usually means the plural form of one word. Under this circumstance, we should use the article “some” and the link verb “are”. Fig. 4 shows the key problems in the process of sentence generation. If we get the element for the video as $w^* = \{\text{dog, grass, walk}\}$, i.e. the object element is “dog”, scene element is “grass”, event element is “walk”, then the corresponding sentence for the video is “a dog is walking on the grass”.

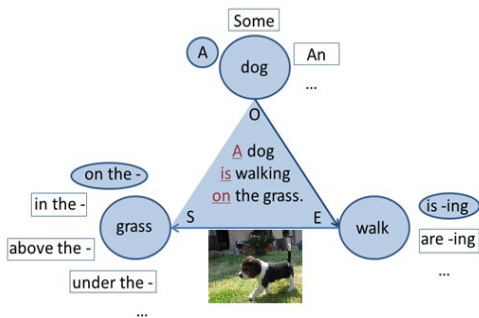


Fig. 4. The key problems in the process of sentence generation.

IV. EXPERIMENT

In this part, we give an introduction on how our experiments are carried out. Our datasets are composed of two parts: image dataset and video dataset. Sentence benchmark and evaluation criteria are also described in detail. At last, the performance of our experiments is shown.

A. Datasets

1) Image dataset

As it is described in Section I, the major obstacle for automatically annotating videos is the insufficiency of labeled training videos due to high labor cost of manual tagging. To overcome this problem, we use another relevant type of media like image as the training data. As we all know that videos are composed of several images, so relevant images can fully demonstrate the content of a video. Images are well-labeled is the other reason why we use them as the training data for image searching.

A part of images in our dataset is downloaded from NUS-WIDE [29], which contains images that are collected from Flickr, there are 425,059 tags associated with these images originally. According to the concept taxonomy of NUS-WIDE, these 81 concepts are categorized into six classes: Events/Activity, Program, Scene/Location, People, Objects and Graphics. Our object image dataset OI includes images on People and Objects. Event image dataset EI includes Events/Activity, Program. Meantime, scene image dataset SI includes images in Scene/Location taxonomy.

In the existing image dataset we know, images are relatively small in scale, so we download some images from the search engine Google and Baidu. As a result, our final dataset include images of NUS-WIDE and images we download from Google or Baidu. The total number of the images in our datasets is 288,270.

2) All elements

In total, we have 128 elements in the Object, Event, Scene and Adjective domains, which are from the NUS-WIDE and the search engine Google and Baidu. The corresponding concept number for Object, Event, Scene and Adjective is 55, 25, 40 and 8 respectively. The total number of images in our dataset is 288,270 with an average of 2,252 images per element approximately. All elements in the four domains are shown in TABLE I.

TABLE I. ALL ELEMENTS IN THE OBJECT, EVENT, SCENE AND ADJECTIVE DOMAINS

OBJECT (55)
alcedoatthis, animal, apple, bear, bird, boat, book, bridge, building, butterfly, car, castle, cat, cherry, clouds, computer, coral, cow, deer, dog, eagle, elk, fish, flag, flowers, food, fox, horse, jeep, lavender, leaf, lotus, military, moon, orange, peacock, person, plane, police, rocks, rose, sailship, sign, statue, strawberry, sun, sunflower, tiger, tower, toy, train, tree, vehicle, whale, zebra
EVENT (25)
dance, drive, eat, earthquake, on fire, fly, hang, jump, land, lie, protest, race, ride, rise, row, run, sail, sit, play soccer, do sports, stand, surf, swim, walk, have a wedding
SCENE (40)
airport, beach, cityscape, room, court, forest, frost, garden, glacier, grass, harbor, highway, house, kitchen, lake, library, mountain, nighttime, ocean, office, plants, playground, railroad, rainbow, reflection, restaurant, road, sand, sea, sky, snow, street, sunset, swimming pool, temple, town, valley, water, waterfall, window
ADJ(8)
blue, bright, dark, gray, green, red, white, yellow

3) Video dataset

Our experiments are conducted on a database of real-world online videos we downloaded from the famous video portal YouTube. There are several online user-generated videos with diversified content. The total duration is about 14 hours with 1,887 shots. Each shot lasts for approximately 25 seconds in average. For each shot a key-frame is extracted. The key-frame has the smallest visual feature distance with the other frames. These videos are downloaded from YouTube with different themes: animals, autos, people, sports and travel.

The numbers and ratios of shots with each theme in the entire video dataset are illustrated in Fig. 5. Video shots with the theme animals, autos, people, sports, travel account for 32.5%, 17.5%, 9%, 26%, and 15% respectively.

Among these 1887 shots, we repeated 10 times on the randomly chosen 50 shots to conduct our experiments and evaluate the performance.

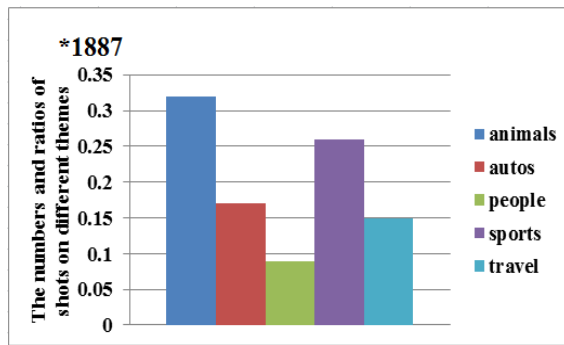


Fig. 5. The number and ratios of video shots about five themes: animals, autos, people, sports, and travel.

B. Visual Features

We extract the visual features of our video frames and images in the image datasets.

A 215-dimensional visual feature vector is applied, which consists of a color feature vector (45-dimensional color moment), and a texture feature vector (170-dimensional HWVP descriptors). The influences of visual features to our sentences generation is also discussed in the following sections.

1) 45-D Color Moment (CM)

Color feature has been proved to be the most GPS-informed feature [30, 31, 39]. Many researchers have dedicated their efforts to improve the image search results with color descriptor [36, 37, 38, 39]. In this paper, it is also used as global feature representation for the image in our method to search the visually similar images. An image is divided into four equal sized blocks and a centralized image with equal-size. For each block, a 9-D color moment is computed, and thus the dimension of color comment for each image is 45. The 9-D color moment of an image segment is utilized, which contains values of mean, standard deviation and skewness of each channel in HSV color space.

2) 170-D Hierarchical Wavelet Packet Descriptor (HWVP)

Texture feature has been shown to work well for texture description of image and for scene categorization and image recognition [32, 39]. We use a hierarchical wavelet packet descriptor (HWVP) [33, 34], a kind of texture feature representation approach, in our approach. A 170-D HWVP descriptor is utilized by setting the decomposition level to three and the wavelet packet basis to DB2.

3) Scale Invariant Feature Transform (SIFT)

The images could be described via the local interest point descriptors given by scale-invariant feature transform (SIFT) [35]. SIFT describes the local gradient distribution of the image [37]. Lowe's method for image feature generation transforms an image into a large collection of feature vectors, each of which is invariant to image translation, scaling, and rotation, partially invariant to illumination changes and robust to local geometric distortion.

First, we randomly sample the SIFT feature points from our image datasets of 288,270 images, and group the SIFT points into C centroids (i.e., the BoW number is C , $C=258,870$) using a hierarchical K-means based approach. For all images in OI, EI, SI, AI and image clusters of testing video shots, we extract their SIFT feature. Then each SIFT point is quantized into one of the C centroids by assigning it to the nearest centers,

although the information loss will be introduced during the quantization [36]. Then, we measure the mean squared distance (MSD) between the BoW histogram of images in OI, EI, SI and each query image extracted from the video shot. The top ranked R ($R=10$) images in each dataset OI, EI, SI are selected as each query image's visual neighbors in each domains. Finally, the corresponding candidate sentence elements are the tags of these visual neighbors.

4) A Coupled Multi-index for Color Name and SIFT

The work proposed by [36, 37] which couples SIFT and color features into a multi-index framework to fuse features in the index-level. For simplicity, we use color-SIFT in our discussions.

For all images in OI, EI, SI, AI and image clusters of testing video shots, we extract the color name and sift features. Then, these two features are quantified to 64-D binary SIFT signature and 22-D binary color name signature respectively, which are combined to form a multi-dimensional inverted index. Next, we measure the Euclidean distance of the multi-dimensional inverted index between the images in OI, EI, SI and each query image extracted from the video shot. The top ranked R ($R=10$) images in each dataset OI, EI, SI are selected as each query image's visual neighbors in each domains. Finally, the corresponding candidate sentence elements are the tags of these visual neighbors.

It is worth mentioned that, although we utilize the global and local feature to annotate video frames, actually, better performance can be achieved by utilizing deep learning features [41]. For simplicity, in this paper, we only utilize the global and local feature for annotation.

C. Sentence Benchmark

For the video shot V we downloaded from YouTube, we manually label it with a sentence S_{bm} to describe its contents. $S_{bm} \supset \{A_{bm}, B_{bm}, C_{bm}\}$. The benchmark sentence contains three main parts: A_{bm} , B_{bm} and C_{bm} . A_{bm} the elements like object, event and scene, i.e. $A_{bm} \supset \{w_o, w_e, w_s\}$. B_{bm} contains the other parts like the article, the link verb and the preposition, i.e. $B_{bm} \supset \{a, l, p\}$. $C_{bm} = w_a$ is the adjective that modifies the scene.

So the benchmark sentence can be demonstrated as follows:

$$S_{bm} \supset \{A_{bm}, B_{bm}, C_{bm}\} = \{a, w_o, l, w_e, p, w_a, w_s\}$$

We have invited 9 volunteers to help us with the labeling task. The benchmark is generated according to the following four rules.

- 1) These 9 volunteers are required to watch and label the video independently.
- 2) In labeling A_{bm} , they are required to choose object, event and scene from our collected elements to describe the video content. And the plural form is allowed. Moreover, if there is no appropriate elements in our dataset, they are asked to use "X" instead.
- 3) In labeling B_{bm} , they are required to select the appropriate article, link verb and preposition to construct the sentence.
- 4) In labeling C_{bm} , they are required to select the appropriate adjective to modify the scene.

Then we make a statistic about the elements in A_{bm} , B_{bm} and C_{bm} given by these 9 volunteers and choose the most frequently used ones as the final benchmark S_{bm} .

D. Criteria of Performance Evaluation

In the performance evaluation process, we generate a sentence S_g by our video annotation approach. Correspondingly, it also consists of three main component, we denote it as $S_g \supset \{A_g, B_g, C_g\}$, where $A_g \supset \{w_o^*, w_e^*, w_s^*\}$, $B_g \supset \{a^*, l^*, p^*\}$, and $C_g = w_a^*$. Then, we compare S_{bm} with S_g and calculate a score to examine how well S_g expresses the video contents.

1) Score_Concept

Both A_{bm} and A_g have three main parts: object, event and scene respectively. If they share only one part of these three parts, the score is 1. If they share two parts, the score is 2. If they share all the parts, the score is 3. If unfortunately, they share no part, the score is 0. This score is calculated to measure the accuracy of concepts, we record it as *Score_Concept* and use SC for simplicity. Thus, a test set with N video shots, we can measure our video sentences generation performances by the percentages of the score categorized as follows:

$$SC(s) = NC(s) / N * 100\%, s = \{0, 1, 2, 3\} \quad (21)$$

where $NC(s), s = \{0, 1, 2, 3\}$ is the number of video shots whose A_{bm} share s parts with the A_g .

2) Score_Sentence

Correspondingly, Both B_{bm} and B_g have three main elements: article, link-verb, preposition respectively. We also define a score to measure how well the sentence is organized. It includes three main parts like the article, the link verb and the preposition. The score 0,1,2,3 is calculated the same with *Score_Concept*. We record it as *Score_Sentence* and use SS for simplicity.

$$SS(s) = NS(s) / N * 100\%, s = \{0, 1, 2, 3\} \quad (22)$$

where $NS(s), s = \{0, 1, 2, 3\}$ is the number of video shots whose B_{bm} share s parts with the B_g .

3) Score_Adj

At last, we get the score of the adjective, which is recorded as *Score_Adj* and use SA for simplicity. Due to the fact that there is only one element in this domain, we have

$$SA = \begin{cases} 1, w_a = w_a^* \\ 0, w_a \neq w_a^* \end{cases} \quad (23)$$

$$SA(s) = NA(s) / N * 100\%, s = \{0, 1\} \quad (24)$$

where $NA(s), s = \{0, 1\}$ is the number of video shots whose C_{bm} share s parts with the C_g .

4) WAP and AP

After getting *Score_Concept*, we use two parameters AP (Average Precision) and WAP (Weighted Average Precision) to demonstrate the performance differences of our experiments.

The definitions of AP and WAP are denoted as follows.

$$AP = \frac{NC(3) + NC(2) + NC(1)}{NC(3) + NC(2) + NC(1) + NC(0)} \quad (25)$$

$$WAP = \frac{NC(3) + 0.8NC(2) + 0.5NC(1)}{NC(3) + NC(2) + NC(1) + NC(0)} \quad (26)$$

As we can see in the definition, the higher AP and WAP are, the better our result is.

E. Performances

Our evaluation focuses on un-constrained online videos. In our experiments, we repeated 10 times to generate sentences for random selected 50 video shots using our four-step approach: video preprocessing, candidate sentence element acquisition, best sentence element selection and sentence generation.

Comparison experiments are conducted to discuss the effectiveness of the weighted scoring algorithm and the correlation graph algorithm. We compare four methods in this part. These four methods are different only in the process of best sentence element selection and sentence generation. In the process of best sentence element selection, we use a weighted scoring algorithm. While in the process of sentence generation, a correlation graph algorithm is used.

The first method denoted as I in Fig. 6 is using the tag of key frame as the final sentence element directly. For the key frame, we acquired a series of candidate sentence elements. In method I, the key frame tag is with the highest occurrence number in candidate sentence elements of each domain (O, E, S, A). The second method IC utilizes the correlation graph algorithm on the key frame tag to choose the best elements. The third method WS is using only the weighted scoring algorithm to choose the best elements. The last method WC utilizes both the weighted scoring algorithm and the correlation graph algorithm. The shots percentage of these four methods under different SC are shown in Fig.6.

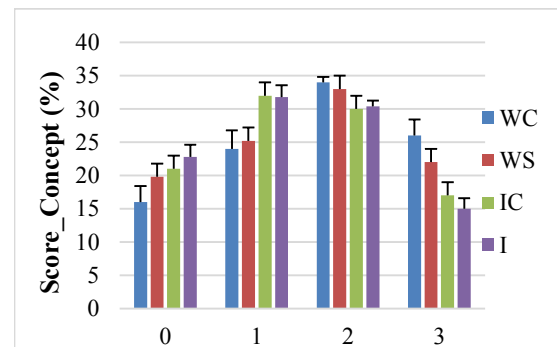


Fig. 6. SC of different video methods: WC, WS, IC, and I.

As we can see in Fig. 6, the percentage of 2 or 3 points in WC is the highest comparing to the left methods, WS is the second highest, IC is the third highest, I is the lowest. While the percentage of 0 and 1 points in WC is the lowest comparing to the other three methods, WS is the second lowest, IC is the third lowest, I is the highest. And this has proved the effectiveness of the weighted scoring algorithm and the correlation graph algorithm.

Besides, AP and WAP of methods WC, WS, IC, I are shown in Table II.

TABLE II. AP AND WAP OF METHODS WC, WS, IC, AND I

	WC	WS	IC	I
AP	0.84	0.79	0.78	0.76
WAP	0.652	0.576	0.573	0.520

In TABLE II, the method I has the lowest AP and WAP, while the method WC has the highest AP and WAP. That's to say, the correlation graph algorithm and the weighted scoring




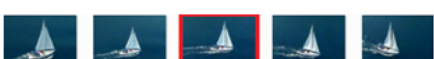


Video snapshots (key frames in the red box)	Benchmark	Our Method	Score	Sentence (benchmark/ generated)
	(dog, walk, grass)	(dog, walk, grass)	3	A dog is walking on the green grass.
	(a, is, on)	(a, is, on)	3	A dog is walking on the green grass.
	(green)	(green)	1	
	(plane, land, airport)	(plane, land, airport)	3	A plane is landing in the bright airport.
	(a, is, in)	(a, is, in)	3	A plane is landing in the gray airport.
	(bright)	(gray)	0	
	(plane, fly, sky)	(plane, fly, sky)	3	A plane is flying in the gray sky.
	(a, is, in)	(a, is, in)	3	A plane is flying in the gray sky.
	(gray)	(gray)	1	
	(boat, sail, ocean)	(whale, sail, ocean)	2	A boat is sailing on the blue ocean.
	(a, is, on)	(a, is, on)	3	A whale is sailing on the blue ocean.
	(blue)	(blue)	1	
	(sun, rise, ocean)	(sun, rise, water)	2	The sun is rising from the ocean.
	(the, is, from)	(a, is, in)	1	A sun is rising in the dark water.
	(X)	(dark)	0	
	(person, riding, X)	(horse, run, road)	0	A person is riding on somewhere.
	(a, is, on)	(a, is, on)	3	A horse is running on the dark road.
	(X)	(dark)	0	

Fig. 7. Examples of our generated sentences. Video snapshots and corresponding benchmarks are also given. The generated sentences and its scores are in the right two columns.

algorithm which are introduced to choose the best elements are indispensable and effective to make an appropriate sentence for our experimental video dataset. Besides, the method WS acquires better performances than the method IC. We can see that the weighted scoring algorithm takes a more important role than the correlation graph algorithm in identifying the best elements, because without good candidate elements, the ranking process of the correlation graph algorithm does not work.

Then we compare our generated sentence with the benchmark to acquire SC, SS and SA to evaluate the performance of WC. We make a statistic about the percentage of video shots that having different scores under the method WC. The scores are shown in TABLE III.

TABLE III. THE PERFORMANCE OF WC

score	0	1	2	3
SC	16%	20%	36%	26%
SS	8%	16%	20%	56%
SA	30%	70%	—	—

As we can see in TABLE III, video shots with the SC 0,1,2,3 account for 16%, 20%, 36%, and 26% respectively. Video shots with the SS 0,1,2,3 account for 8%, 16%, 20%, and 56% respectively. Video shots with the SA 0, 1 account for 30%, 70% respectively. According to our experimental results, most of our generated sentences have more than two right elements and appropriate structure, which demonstrates the effectiveness of

our method WC.

Besides, we have shown some examples of our generated sentences in Fig.7.

F. The influence of parameter R

In the process of candidate sentence element acquisition, we have introduced a parameter R . R is the number of visual neighbors for each image in I^g . In this part, we will analyze the performance of our approach under different R .

First, we set R as 1,5,10 and 20 respectively. Then we implement methods WC, WS, IC and I to see the influence of R . The performance (WAP) comparison of these four methods under different R are shown in TABLE IV.

TABLE IV. THE PERFORMANCE COMPARISON OF WC, WS, IC AND I UNDER DIFFERENT R

WAP	WC	WS	IC	I
R=1	0.507	0.451	0.428	0.362
R=5	0.583	0.537	0.513	0.458
R=10	0.652	0.576	0.573	0.520
R=20	0.617	0.561	0.539	0.483

According to TABLE IV, performances of these four methods differs under different R . With more visual neighbors, more relevant tags may be brought in our recommendation list. However, it may also introduce more noise. We want to find a proper R to achieve the tradeoff between diversity and accuracy.

As we can see in TABLE IV, as R grows from 1 to 10, WAP of WC, WS, IC, I increase as well. These four methods have the

lowest performances when R is 1. Because, in this case only nearest neighbor (the most similar image) is selected and its tags are utilized for annotation, it will introduce limited tags and each tag only appears once which will wipe off the appropriate tags. But when R is 20, their performances cease to increase, for too many visual neighbors bring the noisy tags into the candidate elements. So it has proved that WAP won't increase so much as R grows. Therefore, we set R=10 in our experiments.

G. The influence of weights on images from the query image cluster

In the best sentence element selection process, we propose a weighted scoring algorithm, in which different images from the query image cluster are given different weights. The weights of images near the key frame are given with higher weights. In this part, we will analyze the performance of our approach under different weights. The performance (WAP) comparison of our proposed method under different α, β in Eq. (13) are shown in TABLE V.

TABLE V. WAP OF OUR EXPERIMENTS CONDUCTED ON VIDEO SHOTS WITH DIFFERENT WEIGHTS

WAP	WC
Weights I ($\alpha=0.8, \beta=0.5$)	0.652
Weights II ($\alpha=0, \beta=0$)	0.577
Weights III ($\alpha=1, \beta=1$)	0.612
Weights IV ($\alpha=1, \beta=0$)	0.598
Weights V ($\alpha=0.5, \beta=0.5$)	0.620

According to TABLE V, performances of WC under Weights I ($\alpha=0.8, \beta=0.5$) outperforms other four groups, so we set $\alpha=0.8, \beta=0.5$ in our experiments. The weight III ($\alpha=1, \beta=1$) employs the same weight 1 on each image, which will magnify the impact from the noise elements and swamp the appropriate elements. The Weight II ($\alpha=0, \beta=0$) and Weights V ($\alpha=0.5, \beta=0.5$) assign a same weight to the non-key frame images, which ignores the temporal consistency of the video contents. Besides, Weight IV ($\alpha=1, \beta=0$) take the key frame image and near key-frame images into consideration, which limit the numbers of candidate elements and obtain a lower WAP. Weights I ($\alpha=0.8, \beta=0.5$) take the above weaknesses into consideration and introduce a better trade-off between the candidate element numbers and the noise element numbers.

H. The Influence of Video Theme to Our Approach

The video shots we conduct our experiment WC on have 5 themes: animals, autos, people, sports, and travel. We make experiments to discuss the influence of different themes to our approach. The statistics of SC in different video theme are shown in Fig. 8.

The performances of our approach on video shots with different themes are not the same according to Fig. 8. Our method on video shots with the theme people (c) performs poorly due to the complexity of human's behaviors in the video

shots. Video shots with other themes have satisfactory results to some extent. The SC of the most video shots in other themes is more than 2.

AP and WAP of our experiments conducted on video shots with different themes are shown in TABLE VI.

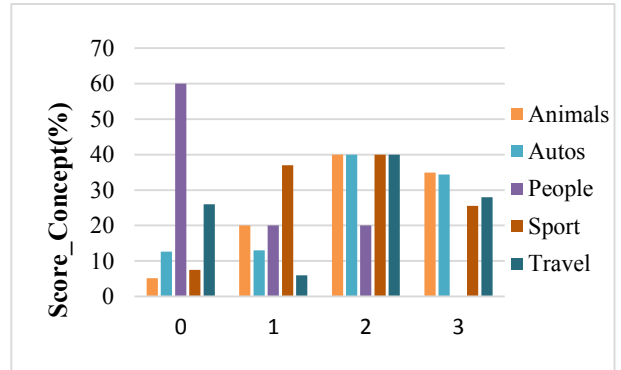


Fig. 8. The statistics of SC in different video theme: animals (a), autos (b), people (c), sports (d) and travel (e).

TABLE VI. AP AND WAP OF OUR EXPERIMENTS CONDUCTED ON VIDEO SHOTS WITH DIFFERENT THEMES

	Animals	Autos	People	Sports	Travel
AP	0.943	0.867	0.4	0.92	0.733
WAP	0.763	0.72	0.26	0.7	0.62

In TABLE VI, video shots with the theme “animals” have the best performance. Its AP is as high as 0.943. Its WAP is 0.763. While video shots with the theme “people” have the lowest performance. Its AP is 0.4 and its WAP is only 0.26. The complexity of human's behaviors in the video shots should be blamed.

I. The Influence of Visual Features

In this paper, we extract 215-D global visual features (45-dimensional color moment, 170-dimensional HWVP descriptors) to describe the content of query image cluster $I^q = \{I_1^q, I_2^q, \dots, I_T^q\}$ and images in our datasets OI, EI, SI and AI.

The process of candidate sentence element acquisition is to find the visual neighbors of every image in the query image cluster $I^q = \{I_1^q, I_2^q, \dots, I_T^q\}$. Then the tags of these visual neighbors are regarded as the candidate sentence element. The principle on acquiring candidate sentence element is all the same when using different features to represent image contents.

Here we give a brief comparison for utilizing different features to find similar images, including the SIFT feature and color-SIFT. After acquiring the candidate sentence element, other steps such as best element selection and sentence generation are all the same with what is explained in Section III. The comparison of using 215-D global feature, SIFT and multi-index for color sift are shown in TABLE VII.

TABLE VII. THE PERFORMANCE COMPARISON OF USING DIFFERENT VISUAL FEATURE: 215-D, SIFT AND COLOR-SIFT

	215-D	SIFT	Color-sift
AP	0.84	0.85	0.86
WAP	0.652	0.697	0.7

As it is shown in TABLE VII, both 215-D global SIFT and Color-sift are effective in our approach. And the performance of using SIFT and color-sift are better than using 215-D global features, for the sift feature is invariant to the image scaling, rotation and color-sift takes the color-name into consideration in addition to sift. However, using SIFT and color-sift are relatively time consuming, so we use the 215-D global features in Section IV.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose an approach for sentence generation from videos. We make a good use of well-labeled image datasets to find sentence related elements. These elements include four main parts: object (the subject of the sentence), event (the action), scene (the place where the action happens), and adjective (modifier of the scene). We can speculate that with more images and accurate tags in our dataset, our experimental performance can be more satisfactory. We have proposed two algorithms to improve the experiment performances. These two algorithms are effective based on our discussions. We also make discussions on how the number of similar images and different selection of visual features influence our results.

However, the corresponding video sentence generation approach can be further improved from following three aspects. First, the structure of our generated sentences is quite simple now. We will dig deep on how to produce more complex sentences by adding more sentence elements and modifiers. What is more, we haven't taken the problem of multiple objects into consideration yet. At last, the scale of our image datasets is relatively small and the number of all our elements is far from adequate. We will collect more images and elements in order to explain more videos.

REFERENCES

- [1] A. Altadmri, A. Ahmed, "A framework for automatic semantic video annotation utilizing similarity and commonsense knowledge bases," in MTA, Springer US, Mar. 2013.
- [2] A. Barbu, A. Bridge, Z. Burchill, D. Coroian, S. Dickinson, S. Fidler, A. Michaux, S. Mussman, N. Siddharth, D. Salvi, L. Schmidt, J. Shanguan, J. M. Siskind, J. Waggoner, S. Wang, J. Wei, Y. Yin, and Z. Zhang, "Video in sentences out," in *Proc. UAI*, 2012, pp. 102–112.
- [3] Y. Yang, Y. Yang, Z. Huang, H. T. Shen, "Transfer tagging from image to video," in *Proc. ACM MM*, 2011.
- [4] K. Yang, X. S. Hua, M. W. H. J. Zhang, "Tag tagging: Towards more descriptive keywords of image content", *IEEE Trans. Multimedia*, vol. 13, no. 4, pp. 662-673, 2011.
- [5] TRECVID, <http://www-nlpir.nist.gov/projects/trevcid/>.
- [6] M. Wang, X. S. Hua, R. Hong, J. Tang, G. J. Qi, Y. Song, "Unified video annotation via multi-graph learning," *IEEE Trans. CSVT*, vol. 19, no. 5, pp.733 -746, 2009.
- [7] M. Wang, X-S.Hua, Y. Song, X. Yuan, S. P. Li, H. J. Zhang, "Automatic video annotation by semi-supervised learning with kernel density estimation," in *Proc.ACM MM*, 2006.
- [8] E. Moxley, T. Mei, X. S. Hua, W. Y. Ma, B. Manjunath, "Automatic video annotation through search and mining," in *Proc. ICME*, 2008.
- [9] J. Tang, X. S. Hua, M. Wang, Z. Gu, G. H. Qi, X. Wu, "Correlative linear neighborhood propagation for video annotation," *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no. 2, pp. 409-416, 2009.
- [10] A. Ulges, C. Schulze, D. Keysers, T. M. Breuel, "Content-based video tagging for online video portals," in *MUSCLE/Image-CLEF Workshop*, 2007
- [11] E. Moxley, M. Tao, B. S. Manjunath, "Video annotation through search and graph reinforcement mining", *IEEE Trans. Multimedia*, vol. 12, no.3, pp. 184-193, 2010.
- [12] S. Huron, P. Isenberg, J. D. Fekete, "PolemicTweet: Video Annotation and Analysis through Tagged Tweets", *Human-Computer Interaction-INTERACT*, pp.135-152, 2013.
- [13] Y. Seok, H. Lee, "Walkietagging: efficient video annotation method based on spoken words for smart devices," 2012 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012.
- [14] T. Mei, X. S. Hua, H. Q. Zhou, S. Li, "Modeling and mining of users' capture intention for home videos," *IEEE Trans. Multimedia*, vol. 9, no.1, pp. 66-77, 2010.
- [15] G.D. Li, Z. Lu, R.C. Hong, X. S. Hua, "In-video product annotation with web information mining," *ACM Trans. on MCCA*, Vol. 8 Issue 4, Nov. 2012.
- [16] Mossi, J. M., Albiol, A., Albiol, A., Oliver, J, "Ground truth annotation of traffic video data", *Multimedia Tools and Applications*, pp.1-14, 2013.
- [17] C. Sun, B. Bao, and C. Xu, "Verb-Object Concepts Image Classification via Hierarchical Nonnegative Graph Embedding," in *Proc. MMM*, 2013, pp.58-69.
- [18] G. Tian, G. L. Guan, Z. Y. Wang, and D. G. Feng, "Annotating images with verbs," in *Proc. ACM MM*, 2012, pp. 1077-1080.
- [19] Y. Ushiku, T. Harada, Y. Kuniyoshi, "Automatic sentence generation from images," in *Proc. ACM MM*, 2011.
- [20] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, D. Forsyth, "Every Picture Tells a Story: Generating Sentences from Images," *Lecture Notes in Computer Science*, vol. 6314, pp 15-29, 2010.
- [21] L.J. Li, F. F. L, "What, where and who? Classifying events by scene and object recognition," in *Proc. IEEE ICCV*, 2007.
- [22] B. Yao, X. Yang, L. Lin, M. Lee, S. C. Zhu, "I2T: Image parsing to text description," in *Proc. IEEE 98*, pp. 1485 -1508, 2010.
- [23] C.C. Tan, Y. G. Jiang, C. W. Ngo, "Towards textually describing complex video contents with audio-visual concept classifiers," in *Proc. ACM MM*, 2011.
- [24] A. Smeulders, M. Worring, S. Santini, A. Gupta, R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. PAMI*, vol. 22, no. 12, pp.1349-1380, 2000.
- [25] X. Wu, W. L. Zhao, and C. W. Ngo, "Towards google challenge: combining contextual and social information for web video categorization," in *Proc. ACM MM*, 2009.
- [26] L. Yang, J. Liu, X. Yang, X. S. Hua, "Multi-modality web video categorization," in *Proc. the international workshop on Workshop on multimedia information retrieval*, pp. 265-274, ACM, 2007.
- [27] X. Yuan, W. Lai, T. Mei, X. S. Hua, X. Q. Wu, S. Li, "Automatic video genre categorization using hierarchical SVM," in *Proc. IEEE Image Processing*, pp. 2905-2908, 2006.
- [28] R. L. Cilibrasi, P. M. B. Vitanyi. "The google similarity distance," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 3, pp. 370-383, 2007.
- [29] T. S.Chua, J.H. Tang, R.C. Hong, H.J. Li, Z.P. Luo, Y.T. Zheng, "NUS-WIDE: a real-world web image database from National University of Singapore," in *Proc. ACM ICIVR*, 2009.
- [30] J. Hays, A. Efros, "IM2GPS: Estimating geographic information from a single image," in *Proc. CVPR*, 2008.
- [31] J. Li, X. Qian, Yuan Yan Tang, L. Yang, and C. Liu, "GPS estimation from users' photos," in *Proc. MMM*, 2013.
- [32] B. S. Manjunath, W. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.18, pp. 837-842, 1996.
- [33] X. Qian, G. Liu, D. Guo, Z. Li, Z. Wang, H. Wang, "Object categorization using hierarchical wavelet packet texture descriptors," in *Proc. ISM*, 2009, pp.44-51.
- [34] X.Qian, D.Guo, X.Hou, Z.Li, H.Wang, G.Liu, "HWVP: Hierarchical wavelet packet descriptors and their applications in scene categorization and semantic concept retrieval," *Multimedia Tools Applcat.*, pp. 1-24, 2012.
- [35] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91-110, 2004.
- [36] L. Zheng, S. Wang, Z. Liu, et al. "Packing and padding: Coupled multi-index for accurate image retrieval," in *Computer Vision and Pattern Recognition (CVPR)*, 2014 IEEE Conference on (pp.1947-1954).
- [37] L. Zheng, S. Wang, Q. Tian, "Coupled binary embedding for large-scale image retrieval," in 2014, *IEEE Trans.Image Processing*.
- [38] C. Wengert, M. Douze, H. Jégou, (2011, November). "Bag-of-colors for improved image search," In *Proceedings of the 19th ACM international conference on Multimedia* (pp. 1437-1440).
- [39] J. Li, X. Qian, Y. Tang, L. Yang, and T. Mei, "GPS estimation for places of interest from social users' uploaded photos", *IEEE Trans. Multimedia* 2013.
- [40] Q. Li, Y. Gu, and X. Qian, "LCMKL: Latent-community and multi-kernel

learning based image annotation”, ACM CIKM 2013, pp.1469-1472.

[41]Y. Gu, X Qian, Q. Li, M. Wang, R. Hong, and Q. Tian, “Image Annotation by Latent Community Detection and Multi-Kernel Learning,” IEEE Trans. Image Processing, 2015.

[42]G.-J. Qi, M.-H. Tsai, S.-F. Tsai, L. Cao, T. Huang. “Web-Scale Multimedia Information Networks,” in Proceedings of the IEEE (P IEEE), Volume 100, Issue 9, 2012.

[43]G.-J. Qi, C. Aggarwal, Q. Tian, J. Heng, T. Huang. “Exploring Context and Content Links in Social Media: A Latent Space Method,” in IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI), 2011.

[44]X. Qian, X. Hua, Y. Tang, and T. Mei, “Social Image Tagging with Diverse Semantics”, IEEE Trans. Cybernetics, vol.44, no.12, 2014, pp.2493-2508.

[45]X. Yang, X. Qian, and Y. Xue, “Scalable Mobile Image Retrieval by Exploring Contextual Saliency,” IEEE Trans. Image Processing, vol.24, no.6, 2015, pp.1709-1721.



Xueming Qian (M’10) received the B.S. and M.S. degrees in Xi’an University of Technology, Xi’an, China, in 1999 and 2004, respectively, and the Ph.D. degree in the School of Electronics and Information Engineering, Xi’an Jiaotong University, Xi’an, China, in 2008, after that he was an assistant professor. He was an associate professor from Nov. 2011 to March 2014, and now he was

a full professor. He was awarded Microsoft fellowship in 2006. He was awarded outstanding doctoral dissertations of Xi’an Jiaotong University and Shaanxi Province in 2010 and 2011 respectively. He is the director of SMILES LAB. He was a visit scholar at Microsoft research Asia from Aug. 2010 to March 2011. His research interests include social media big data mining and search. His research is supported by NSFC, Microsoft Research, and MOST.



Xiaoxiao Liu received the B.E. degree from the Xi’an University of Post and Telecommunications, Xi’an, China, in 2011. She is currently sophomore postgraduate with the School of Electronics and Information Engineering in Xi’an Jiaotong University, Xi’an, China. Now she is a MSD student at SMILES LAB.

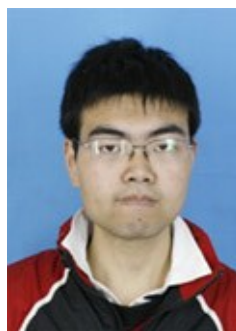


Xiang Ma received the B.S. degree in North China Electric Power University, Beijing, China, in 1999, the Ph. D. degree from Xi’an Jiaotong University, China, in 2011. From 2010 to 2011, he was an exchange Ph.D. student supported by EU with the Image Processing and Interpretation Laboratory, Ghent University, Belgium. He is currently an associate

professor with School of Information Engineering, Chang’an University, Xi’an, China. His research interests include image and video processing.



Dan Lu received the B.S. degree from the Chang’an University, Xi’an, China, in 2013. She is currently sophomore postgraduate with the School of Electronics and Information Engineering in Xi’an Jiaotong University, Xi’an, China. Now she is a MSD student at SMILES LAB.



Chenyang Xu is pursuing his B.S degree in School of Electronic and Information Engineering, Xi’an Jiaotong University. His research interests include image and video pattern recognition.