ShotTagger: Tag Location for Internet Videos

Guangda Li[1,2], Meng Wang[2], Yan-Tao Zheng[2],Haojie Li[3], Zheng-Jun Zha[2], Tat-Seng Chua[2]
NUS Graduate School for Integrative Sciences and Engineering[1]
School of Computing,National University of Singapore[2]
Institute for Infocomm Research[2],Dalian University of Technology[3]
g0701808@nus.edu.sg, yzheng@i2r.a-star.edu.sg {wangm,chuats}@comp.nus.edu.sg

ABSTRACT

Social video sharing websites allow users to annotate videos with descriptive keywords called tags, which greatly facilitate video search and browsing. However, many tags only describe part of the video content, without any temporal indication on when the tag actually appears. Currently, there is very little research on automatically assigning tags to shot-level segments of a video. In this paper, we leverage user's tags as a source to analyze the content within the video and develop a novel system named *ShotTagger* to assign tags at the shot level. There are two steps to accomplish the location of tags at shot level. The first is to estimate the distribution of tags within the video, which is based on a multiple instance learning framework. The second is to perform the semantic correlation of a tag with other tags in a video in an optimization framework and impose the temporal smoothness across adjacent video shots to refine the tagging results at shot level. We present different applications to demonstrate the usefulness of the tag location scheme in searching, and browsing of videos. A series of experiments conducted on a set of Youtube videos has demonstrated the feasibility and effectiveness of our approach.

Categories and Subject Descriptors

H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems—*Video*

General Terms

Algorithms, Design, Experimentation

Keywords

Internet video tagging, tag-based video search, tag-based video browsing

1. INTRODUCTION

In recent years, the modern Web 2.0 activities and contents have pervaded on the Internet. Community contributed video collections on the web are growing in an explosive rate, such as the YouTube archive [3]. These media repositories not only allow users to upload their videos but also encourage them to annotate the videos with descriptive words called tags. Tags provide description of video content, and greatly facilitate the categorization, sharing and search of videos. However, even the tags are provided for a whole video, they may only describe a small part of the video content. As a result, when looking for video information via tags, users are often bewildered by the vast quantity of seemingly unrelated videos returned through video search engines. The users usually have to painstakingly browse through each video to find the interesting parts. Figure 1 illustrates the snapshot of an example. A video is tagged with "plane" and "crash", but the event "plane crash" only happens at the very end of the video. Therefore, if we can decide which shots only annotated with "airplane crash", then users will be able to browse the tag-related highlights easily.

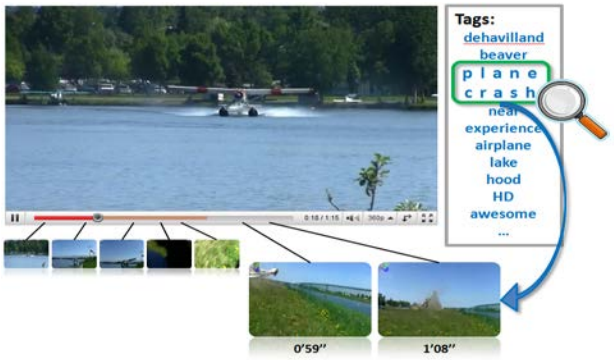

Figure 1: A video and its associated tag list from YouTube. This video is tagged with "plane" and "crash", but the event "plane crash" only happens at 1'09" towards the end of the video.

Thus a system that can automatically assign meaningful tags at the shot level is highly desired. Recently Google has emphasized the significance of temporal localization of video-level labels, by highlighting it as an important challenge in multimedia research [1]. In this work, we introduce a scheme named *ShotTagger*, which accomplishes the tag location task in two steps. The first is to to estimate the distribution of tags within the given video based on a multiple instance learning (MIL) framework. The existence of a video-level tag indicates that the video should contain visual content relating to the tag. Following the framework of multiple instance learning, videos are regarded as bags and shots are regarded as instances. The bag (video) is positive

ICMR'11, April 17–20, Trento, Italy.

only if at least one of the shots is relevant with respect to the intended tag. The task is then cast as one that finds positive instances (shots) in positive bags (videos). However, some video-level tags may be miss-labeled and they are not relevant to any shot within the video. This contradicts the MIL assumption that a positive bag must have at least one positive instance. To eliminate the noise video-level tags, we incorporate the semantic correlation of tags into the optimization scheme. The overall optimization process is referred to as the **context-aware multiple instance learning(CA-MIL)**. After that, the temporal smoothness across adjacent video shots is explored to refine the tag location results. The temporal consistency, which means that contiguous video contents are semantically close with high probability, is an important property of videos as compared to still images.

The National Institute of Standards and Technology (NIST) has established "high-level feature extraction" as a task in TREC video retrieval evaluation (TRECVID) from 2002 [2]. The "high-level feature extraction" task (also referred to "video concept annotation") is actually the tagging of video shots with a predefined set of labels. Extensive research efforts have been dedicated to this task [19] [18] [17] [32]. However, shot-level tagging is different from TRECVID styled "high level feature extraction" in several ways: first, the above methods improve the performance by incorporating more complicated models trained by manual annotation for a large video archive, whereas in this work, we utilize the tags provided by users through social tagging which tend to be more cryptic, ambiguous and noisy; second, TRECVID is mainly designed for narrow domain such as news video, but not for general domain, such as web-based videos.

The main contributions of this paper are as follows. First, we develop a fully automated system that locates the temporal positions of tags in videos at shot level in an unsupervised manner. To the best of our knowledge, this is the first attempt to explore the location of tags for general web-based videos. The context of tags and the consistency of contiguous video shots are both investigated in our learning framework. Second, we design a novel application which is to be supported by the *ShotTagger* scheme.

The rest of this paper is organized as follows. In Section 2 we briefly review related work. Section 3 describes the *ShotTagger* scheme and the details of the algorithms. In Section 4, we conduct evaluation on the performance of tag location. In Section 5, we describe the application that is built upon the ShotTagger scheme. Finally, we offer some concluding remarks.

2. RELATED WORK

In this section, we briefly review research on automatic video tagging, video feature representation, and multiple instance learning.

2.1 Video Tagging

Millions of videos are available on the web with rich text metadata, such as title, comments and tags. Therefore, the multimedia research community has extended the interest to video search and annotation in constrained domain, such as news video, to unconstraint internet video analysis. For example, Damian et al. [4] proposed a system to classify web videos into predefined concept hierarchical categories mainly using visual features. Wu et al. [29] proposed a system to categorize web videos into different genres mainly using video metadata information and social network information. Yang et al. [30] described a genre (such as sports and music) classification system for YouTube videos. They demonstrated that encouraging results can be achieved by exploring multiple modalities. The above efforts focus on categorizing the overall video into different video genres or semantic categories, but not assigning tags at shot level.

Some recent efforts have been made on enriching or exploring the tag information of web videos. Siersdorfer et al. [25] described a scheme that enriches YouTube videos' tag information by exploring their redundancy, such as overlapping or duplicated content. They built a graph for a set of videos, and tags from redundant videos are propagated to the target video through the graph structures. Ulges et al. [28] described a scheme that employs online video portal as a data source for learning concept detector. It treats videos tagged with the target concept as positive training samples and others as negative samples. Our work well complements these research efforts. For example, if tags are located on shots with satisfying accuracy, then better results can be achieved by training concept detectors using shot-level samples and tag propagation can also be performed on a finer graph structure. Ulges et al. [27] also described a similar scheme to ours. They identified the relevant keyframes of a weakly labeled video for training concept detectors using a generative model. We, on the other hand, adopt a discriminative model and exploit contextual knowledge and video structure information for better tag location performance.

2.2 Feature Representation of Internet Video

For feature representation of internet videos, there are mainly three approaches. The first is from visual content point of view. For example, Schindler et al. [7] explored the representation of a video at keyframes level as a "bag of words" using various combinations of spatial and temporal descriptors. They tested several combinations of interest points and descriptors on the YouTube sports dataset and a general 15 categories YouTube dataset using 1-nearest neighbor and centroid-based document classification techniques. They demonstrated 42.7% and 26.9% recognition performance respectively. The second is the combination of visual and audio contents for video concept classification. For example, Jiang et al. [8] extracted short-term region track associated with regional visual features and background audio features to represent videos. They then constructed the codebooks of these features using Multiple Instance Learning.

2.3 Multiple Instance Learning

Multiple Instance Learning (MIL) is a technique to tackle the problems that the label information of training data is incomplete [15], [21]. In a typical supervised learning task, every training instance is associated with a label, but in MIL the labels are only assigned to bags of instances. There has been much work on applying MIL to object-based image retrieval problems [22] [31]. These methods model images as bags and image regions as instances. Many algorithms are proposed to solve the above problem, such as Diverse Density [15] and EM-DD [21]. Here we also apply the MIL approach for our tag location task, treating a video as a bag and each shot is an instance. We extend the multi-instance logistic regression (MILR) [23] [24] and further incorporate

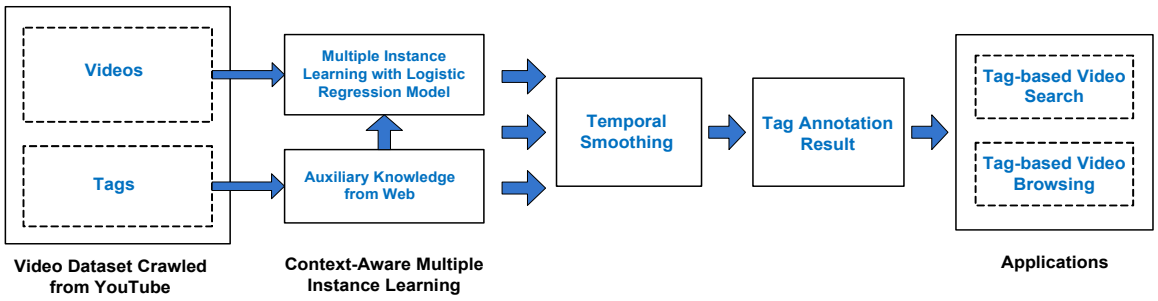

Figure 2: Overall framework of *ShotTaggger*

contextual information to improve performance.

3. APPROACH

In this section, we present the proposed framework in details. First we introduce the whole scheme and then we describe how to find tag-specific video segmentation based on a modified multiple instance learning scheme. Finally, we explore the temporal smoothness of video data to refine the results.

3.1 System Overview

Figure 2 illustrates the general flow of the tag location scheme. First, videos associated with their metadata are collected from YouTube by issuing different queries. The video is split into shots and the corresponding features are extracted for each shot. Second, we employ bag-instance paradigm to model the shots and the video. Since user-provided tags are imperfect and ambiguous, the semantic correlation of co-occurring tags is estimated and then incorporated into the overall optimization procedure, which is called the context-aware multiple instance learning. Third, the annotation results are temporally smoothed by imposing the consistency across shots. The whole process is automated and do not need any manually labeled training data. Finally, the scenario of locating a tag's occurrences in the shots within a video can be integrated into other applications, such as tag-based video search, and tag-based video browsing.

3.2 Locating Tags on Shots

3.2.1 Bag-Instance Modeling of Video

As previously mentioned, videos and shots are regarded as bags and instances, respectively. When a tag appears in a video, there is a relatively high probability that the target concept will appear in some parts of the video. The task is to learn which shot within a positive video is corresponding to the target tag. We extend the multiple-instance logistic regression (MILR) [23] [24] to model the relationship between videos and their shots. The target is therefore to estimate the conditional probability of a shot having a certain tag t given the content of the shot. We denote this conditional probability as $P_t(y_{ij} = 1|f_{ij})$. f_{ij} represents the feature vector of the shot i within the video j. In MILR framework, logistic regression is used to model the conditional probability that a shot contains the tag t:

$$S_{ij} = P_t(y_{ij} = 1|f_{ij}) = \frac{1}{1 + \exp(-(\sum_{n=1}^{D} w_n \cdot f_{ij,n} + b))} \quad (1)$$

where $f_{ij,n}$ is the nth dimension of the feature vector of the shot and w_n is the nth dimension of the weight vector W associated with the feature vectors. The parameter b is a bias term. The unknown parameter pair (W, b) needs to be estimated. For each video, $softmax$ function is used to combine these conditional probabilities for shots into a conditional probability for a video:

$$V_i = P_t(y_i = 1|f_i) = \frac{\sum_{j=1}^{l} S_{ij} \exp(\alpha S_{ij})}{\sum_{j=1}^{l} \exp(\alpha S_{ij})} \quad (2)$$

where $P_t(y_i = 1|f_i)$ is the conditional probability of a video tagged by a certain tag t given the content of this video, and α is a constant that determines the extent to which $softmax$ approximates a hard max function. The above equations (1) and (2) represent THE smooth functions of (W, b). This parameter pair can be determined by an optimization procedure toward minimizing an object function, such as the squared error function:

$$E_t(w_n, b) = \frac{1}{2} \sum_{i=1}^{N} (Y_i - V_i)^2 \quad (3)$$

where $E_t(W, b)$ is the value that needs to be minimized, and Y_i indicates the occurrence of tag t for video j, which is 0 or 1. After the parameter pair (W, b) is estimated, $P_t(y_{ij} = 1|f_{ij})$ can be calculated based on Eq.2.

3.2.2 Incorporating External Knowledge

User-provided tags are imperfect and ambiguous and thus the tags labeled on a video are not necessarily relevant to certain shot within the video. If we force all videos labeled with a specific tag to be positive, the miss-labeling will hurt the multi-instance model. To overcome this problem, we incorporate external knowledge of co-occurring tags of a video as optimization constraints to eliminate the effect of Quasi-Positive Bag. Given a target tag in a video, this tag is likely to be relevant to the content of the video if there are other tags describing this video that are semantically related to this tag. The semantic relatedness between the target tag and the other tags in the same video is measured using the co-occurrence statistics obtained from the external source, in this case the web. Specifically, we calculate the semantic relatedness β of the target tag as the average of similarity with all the other tags in the video, as measured by the web statistics.

To calculate the tag set similarity for a video, the first step is to use a search engine to retrieve candidate snippets(the short summaries of the retrieved documents) by querying each tag separately. We use Google as our search engine. The top z snippets for each tag are retrieved, which are used as context for each tag. Second, the similarity of two

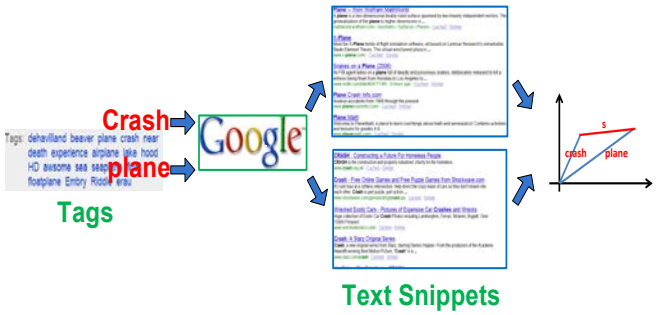

Figure 3: Calculating the tag relatedness using Google Search

tags is calculated as the *cosine* similarity [14] between the corresponding snippets. The tag set similarity is calculated by averaging all the similarity between the target tag and all the other tags for a video. The process is illustrated in figure 3. Of course more complicated tag relatedness measurements can be explored, such as the work done in [26]; we use the cosine similarity for simplicity. In our implementation, a stop words list is utilized to remove the most functional tags such as *I, i, he, her, the, etc.* before searching for snippets.

Since β is a constraint to reflect the sparseness of true positive bags, we treat β as a weight for videos with a target tag. The objective function is modified as follows:

$$E_t(W,b) = \frac{1}{2}\sum_{i=1}^{N}\beta(Y_i - V_i)^2 \quad (4)$$

The idea behind this modification is that videos with higher possibility of correct tag will contribute more to the optimization procedures than videos with lower possibility of correct tag. Thus, the effect of Quasi-Positive Bag is well contained in the optimization procedure. We estimate the parameter $E_t(W,b)$ using gradient-based optimization methods. After the parameter is estimated, the confidence of locating a tag to shots can be calculated based on the Eq.2. The overall process is referred to as a context-aware multiple instance learning with logistic regression (CA-MILR) method.

3.2.3 *Iterative Optimization Procedures*

We estimate the parameter pair using the gradient-based optimization methods. The gradient is a vector whose components are the partial derivatives of the squared error objective function(the gradient function is given in Appendix). For gradient-based optimization methods, it can take many iterations to converge towards a local minimum. So in practice, we update the increment μ according to the directions yielded by each iteration to speed up the iterative process. As long as the iteration process is in a right direction, the increment μ is doubled; otherwise, the increment μ is halved. The detailed iteration process is as given in algorithm 1.

3.3 Temporal Smoothing of Shot-Tag Annotation

The temporal consistency, which means that contiguous video content are semantically close with high probability, is an important property of videos as compared to the still images. Thus adjacent shots are usually semantically similar and they tend to be tagged with the same tag. We therefore utilize temporal smoothing to smooth the results of shot-level tagging. Here, we simply use the nearest neighbor to update the relevance score S_ij of a certain shot as:

$$S_{ij}{}' = \frac{(1-\gamma)}{2}(S_{i,j-1} + S_{i,j+1}) + \gamma S_{ij} \quad (5)$$

where S_{ij-1} and S_{ij+1} are the relevance score of the adjacent shots. In our experiments, we set γ to 0.6. Of course, other complicated smoothing methods like HMM model [9] can also be applied to smooth the predicted relevance score of the shot-level tagging, but we use the nearest neighbor method here due to its simplicity. After we obtain the relevance score of each shots, which is between $[0,1]$, we employ an empirically-set threshold 0.7 to decide the predicted label.

Algorithm 1 Calculate Object function $E_t(W,b)$

Require: Initialize $W = [w_1, w_2, w_3, w_j, b]$ and $\mu = \mu_0$
  while $E_t(W,b)$ not Convergence **do**
    Update $E_t(W,b) = \frac{1}{2}\sum_{i=1}^{N}\beta(Y_i - V_i)^2$
    if $E_t{}^{new}(W,b) < E_t(W,b)$ **then**
      $\mu^{new} = 2 * \mu^{old}$
    else
      $\mu^{new} = 1/2 * \mu^{old}$
    end if
    $w_n^{new} = w_n^{old} - \mu * \delta w$
    $b^{new} = b^{old} - \mu * \delta b$
    Update S_{ij} and V_i
  end while

4. PERFORMANCE EVALUATION

In this section, we evaluate the shot-level tagging framework described above in term of annotation and tag-based shot retrieval, based on a pilot internet video dataset of more than 1000 videos (Section 4.1). We will first show the accuracy for shot-level tagging in video (Section 4.2), and then show the tag-based video retrieval performance for tag-based shots search (Section 4.3).

4.1 Constructing Shot-Tag Video Database

To evaluate our *ShotTagger* system in an experimental evaluation, we assemble a large video dataset by issuing different queries over YouTube from Oct 2009 to Feb 2010, through YouTube API. The queries we selected for downloading are based on the LSCOM annotated events [10] and some of the "high level" concepts which fall into several categories including events (e.g. airplane crash,), scenes (e.g. basketball match), or particular objects (e.g. bear). We pick the most frequent tags from each query. However, some of the semantic meaningful events cannot be conveyed by only a single word, such as the "car crash" which consists of two words "car" and "crash"; or "street battle" that consists of two words "street" and "battle". Therefore we also evaluate this two-word tag situation. We also conduct the manual checking on whether the tag occurs in the shots. Two people are involved in the manual annotation process. Shots with conflicting annotation label were re-checked until the annotators reached an agreement. After removing some tags with only very few videos, 23 tags and their corresponding morphological affixes are selected. We organized these tags in Table 1. Overall, there are 1,224 videos and 49,154 shots. The positive and negative shots given each tag in all is 15025 and 34129 respectively. Despite the limited testing concepts,

Table 1: Shot annotation accuracy using different methods. PosNum:the number of positive Shots; NegNum: the number of negative Shots; ZeroR: the percentage of positive shots; MILR: multiple instance learning with logistic regression; MILR–TS: multiple instance learning with logistic regression and temporal smoothing; CA-MILR: context-aware multiple instance learning with logistic regression. CA–MILR–TS: context-aware multiple instance learning with logistic regression and temporal smoothing.

ConceptName	PosNum	NegNum	ZeroR	MILR	MILR-TS	CA-MILR	CA-MILR-TS
airplane_crash	164	3438	0.05	0.06	0.07	0.07	**0.07**
airplane_flying	380	1710	0.18	0.29	0.37	0.32	**0.42**
airplane_landing	43	662	0.06	0.08	0.06	0.07	**0.08**
airplane_takeoff	88	549	0.14	0.19	0.2	0.2	**0.21**
basketball	1521	901	0.63	0.71	0.82	**0.9**	0.87
bear	782	1306	0.37	0.44	0.45	0.43	0.46
car_crash	284	1397	0.17	0.18	0.16	0.21	0.21
cheering	1221	1212	0.5	0.73	0.83	0.78	**0.9**
children	280	3241	0.08	0.07	0.06	0.05	**0.08**
cooking	2467	1711	0.59	0.65	0.67	0.67	**0.7**
dancing	1048	1688	0.38	0.49	0.5	0.48	**0.56**
helicopter_hovering	345	365	0.49	0.51	**0.56**	0.52	0.52
interview	580	1253	**0.32**	0.24	0.23	0.24	0.24
laughing	232	1077	0.17	0.26	0.35	0.25	**0.42**
people_hugging	302	1047	0.22	0.22	0.18	0.2	**0.29**
people_kissing	153	699	0.18	0.21	0.2	0.23	**0.23**
people_marching	415	519	0.44	0.44	0.48	0.45	**0.45**
riot	504	3817	**0.12**	0.09	0.07	0.07	0.07
running	390	2719	0.13	0.14	0.14	0.14	**0.21**
shooting	174	1669	0.09	0.13	0.13	0.17	**0.17**
soccer	2513	1080	0.7	0.79	0.84	0.85	**0.87**
walking	117	1884	0.06	0.06	0.05	0.07	**0.07**
wedding_ceremony	1022	185	0.85	0.89	0.9	0.88	**0.91**
average	**653**	**1484**	**0.29**	**0.34**	**0.36**	**0.36**	**0.39**

our method is designed to be generally applicable to a wide range of contents.

4.2 Performance of Shot-Tag Annotation

In this experiment, we evaluate the performance of the shot-level tagging framework described in section 3 on our pilot dataset. For feature representation, we take shot as the basic content unit to process videos. Videos are split into shots and then one key frame is extracted from each shot using the shot boundary detection method described in [6]. For each keyframe, 128 dimension SIFT feature [13] is extracted; the SIFT feature is then quantized into 1111 dimensions bag of words (BoW) feature using hierarchy clustering method [20]. PCA technique is utilized to reduce the dimension of BoW feature to 200 dimensions.

The goal of our framework is to annotate the target tag at the shot level, which is a binary classification task without needing to know the ranking position of the shots. Thus, the average precision (AP) is not suitable since this metric emphasizes the relevant degree of each relevant document in the ranked sequence. Instead, we use accuracy to measure the performance of locating tags into video. Given a target tag, the accuracy is computed as the proportion of the total the number of shots correctly tagged to the total number of correct shots with the tag in the ground truth. We compare the following methods:

- **Baseline**: we use the modified zero attribute rule (ZeroR) as the baseline method. Traditional ZeroR always returns the most frequent class in the training set. However, since our target is to find the right tagged shots, we treat the all shots as positive and then check the precision rate.
- **MILR**: we utilize multiple instance learning with logistic regression to estimate the distribution of tag, without incorporating the co-occurring tag statistics, as described in Section 3.2.1.
- **MILR–TS**: the MILR framework incorporated with temporal consistency across contiguous shots.
- **CA–MILR**: the context-aware multiple instance learning with logistic regression framework, with the incorporating of co-occurring tag statistics, as described in Section 3.2.2.
- **CA–MILR–TS**: CA–MILR with the enforcement of temporal consistency across contiguous shots.

The results of the using the above methods are shown in Table 1. From the results we can see that:

- The performance of the classification using MILR is much better than the baseline method. This means that the proposed model can really improve the automatic shot-level tagging effectively.
- The incorporation of temporal consistency across shots improves the results of shot-level tagging. Table 2 reveal that MILR-TS has improvement of about 6% over MILR, while CA-MILR-TS registers 8% improvement over CA-MILR.

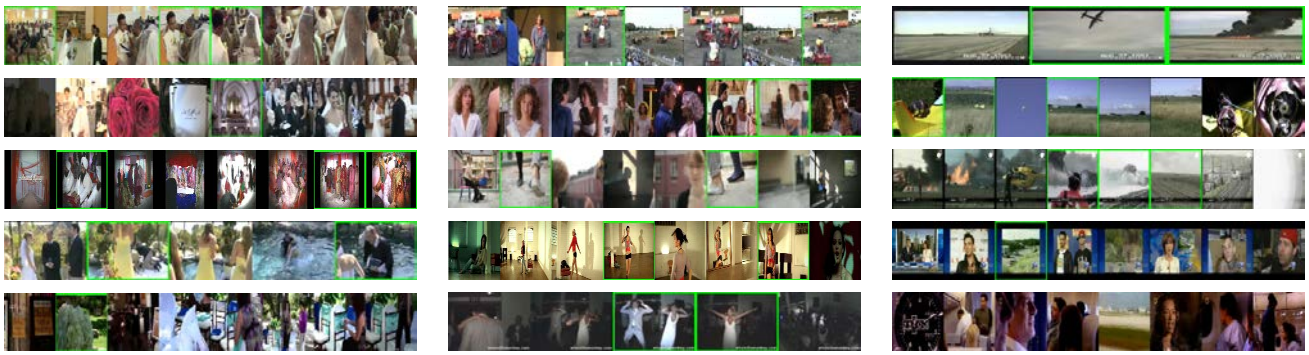

Figure 4: Examples of tag annotation for different tags (left: dance; middle: wedding and ceremony; right: airplane and crash). The threshold to classify positive and negative shots is set as 0.6. The shots in green rectangle are positive with regard to the tag.

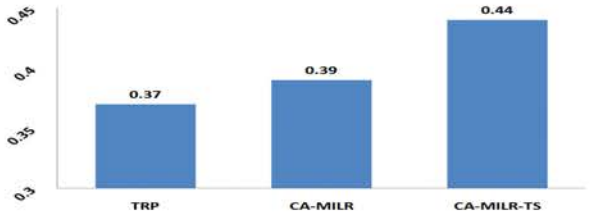

Figure 5: Mean average precision (MAP) for *ShotTagger* system using different methods. TRP: using textual search first and then pick the shots contiguously.

- CA–MILR outperforms the MILR method by 6%, demonstrating the usefulness of the incorporating co-occurring tag statistics as a constraint to find the relevant shots. In fact, with the incorporation of external knowledge from the web, the performance on 18 concept tags was increased, except for "bear", "children', "dancing", "laughing", "riot", "hugging", "airplane and landing', and "wedding and ceremony".

Figure 4 illustrates the results of shot-level tagging for different tags (left: "airplane" and "crash"; middle: "wedding" and "ceremony"; right: "dance"). It is clear that the contiguous shots have a tendency to have the same semantic meaning.

4.3 Performance of Tag-based Shot Retrieval

Currently, YouTube provides four ranking options for video search. The first is "relevance", which orders the videos based on the similarity of query to video's metadata, such as the title, description and tags. The second is "upload data", which orders the videos based on their uploading time. The other two are "view count" and "rating", which orders the videos by the number of clicks on video and user's rating. These four ranking methods are not related to video contents. Thus tag-based video search can complement existing video search.

Here we show how tag-based video search works. Given a query tag, we estimate the relevance levels of shots based on the automated shot annotation results. Thus the more relevant the shots is to a query, the higher the ranking score of the shots. The average precision (AP) and mean average precision (MAP) are used as performance measure. AP is a metric emphasizing the relevant degree of each relevant document in a ranking list. MAP is calculated by averaging the APs across all concepts. To evaluate the proposed tag-based video ranking scheme, we conduct experiments on the following three methods:

- **Text-based video search**: when user inputs a query, the videos are returned based on textual relevance first, and then each shot from videos are presented to the user in chronological order.This serves as the baseline for the comparison with our framework.
- **CA–MILR**: the context-aware multiple instance learning with logistic regression framework, with the incorporating of co-occurring tag statistics, as described in Section 3.2.2.
- **CA–MILR–TS**: CA–MILR with the enforcement of temporal consistency across contiguous

Due to the space limitation, we only report MAP for the above methods. The comparison are shown in figure 5. From the results we can see that:

- The performance of the tag-based shot search using CA-MILR is better than the textual-based video search method, by 5%. This means that the proposed model can improve the searching tag related shots.
- The incorporation of temporal consistency across shots improve the MAP of tag-based shot retrieval. Figure 5 reveals that CA–MILR–TS achieves the best MAP of 0.44, which is has improvement about 13% over CA-MILR method.

5. TAG-BASED VIDEO BROWSING

Shot-level tagging can potentially impact a wide-range of applications. We present only one application here, tag-based video browsing, which is on top of the shot-level tagging framework.

Tag-based video browsing is the natural way to present the tag-related shots to the user. User can immediately select the video content that related to a tag or more than one tag, which enables users to browse the content efficiently. The research problems here are what-to present and how-to present. Yang et al. [11] presented a content-based smart video player. User can browse the video content with the filmstrip view through the keyframes extracted by on-the-fly video parsing techniques. Cheng el al. [5] proposed an adaptive fast-forwarding video player which can adaptively

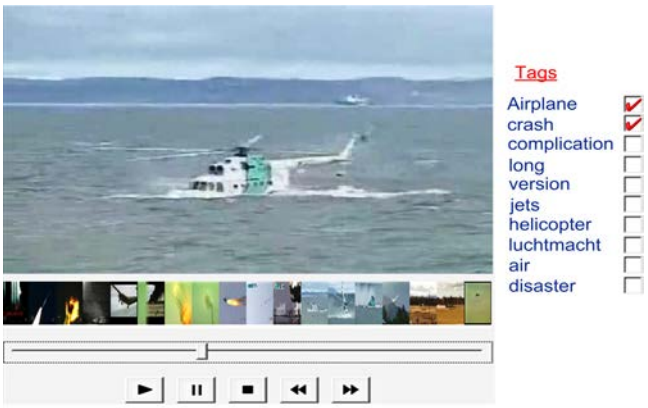

Figure 6: Tag-based video browsing.

adjust playback speed based on video content and predefined rules. User can click a specific key-frame to access the corresponding video content in both ways.

Learning from existing approaches, our content-based video browsing is tag-based video browsing, in which user can select the tag-related content from the presented filmstrip. For example, in Figure 6, a user selects two tags "airplane" and "crash", and system automatically presents a list of relevant keyframes in a way of filmstrip, which is between video and the time axis.This is based on the pre-processed shot-level tagging framework. By clicking on the keyframe shown in filmstrip, user can easily go to the corresponding shot by fast forwarding the video. He or she still has the option to select other tags and then view the relevant filmstrip. Thus, user can easily find the tag-related shots within the video sequence.

User evaluation is conducted to validate the usability of tag-based video browsing. 10 participants were asked to look for a certain event (indicate by the selected tag) in videos using the tag-based video browsing player. They are also asked to find the event by just watching the video. The expectation is that evaluators can find all the relevant information in an easy way. They were asked to give scores ranging from 1 to 5 based on their satisfaction, with higher score meaning better satisfaction. For each tag, We define the satisfaction as the convenience of finding the relevant information that evaluators should answered in the questionnaires. The comparison of average satisfaction for each tag is shown in Figure 7.Overall, the result shows that tag-based browsing scores more than just watching the video. The evaluation confirms with our expectation that the tag-based video browsing of web videos can effectively speed up the process of grasping the video fragment according to a certain tag.

6. CONCLUSION AND FUTURE WORK

In this paper, we explored how to leverage user's tags as a source to understand the content of video and develop a novel system: *ShotTagger*. We used a combination of context and content based method to annotate the shots corresponding to the same tag within an internet video. The method which is named context-aware multiple instance learning, is fully unsupervised and do not need any training data. Since the user provided tags are imperfect and ambiguous, we incorporated external knowledge from co-occurring tags of target tag as constraints. Besides, temporal smoothing is also imposed to refine the annotation result

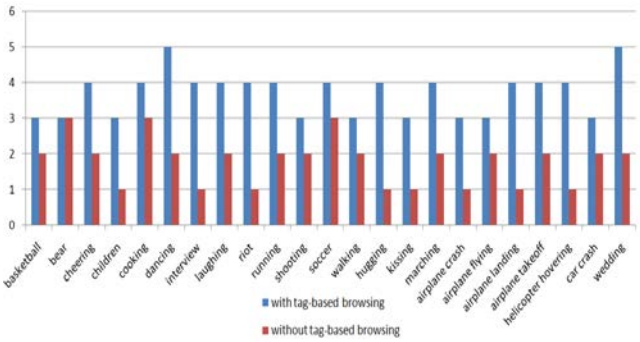

Figure 7: User subjective convenience score of using tag-based browsing and without using it.

to impose the consistency across shots. A series of experiments, including tag annotation and tag-based shot search are conducted using the videos from YouTube archive. The results showed the feasibility and effectiveness of our approach. In addition, we presented how locating tags into video can be built into a tag-based video browsing.

For future works, we will explore more advanced video shots feature representation techniques, such as the work done in [12], and test them with more tags [16]. On the application side, we will incorporate video summarization and video suggestion [33] into our system.

7. ACKNOWLEDGEMENT

This research was supported by NRF (National Research Foundation of Singapore) Research Grant 252-300-001-490 under the NExT Search Center.

8. REFERENCES

[1] Google multimedia research interest: *http://googleresearch.blogspot.com/2009/12/research-areas-of-interest-multimedia.html/*.

[2] Trec video retrieval evaluation: *http://www-nlpir.nist.gov/projects/trecvid/*.

[3] Youtube video: *http://www.youtube.com/*.

[4] D. Borth, J. Hees, M. Koch, A. Ulges, C. Schulze, T. Breuel, and R. Paredes. Tubefiler: an automatic web video categorizer. In *MM '09: Proceedings of the seventeen ACM international conference on Multimedia*, pages 1111–1112, 2009.

[5] K.-Y. Cheng, S.-J. Luo, B.-Y. Chen, and H.-H. Chu. Smartplayer: user-centric video fast-forwarding. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, 2009.

[6] H. Feng, A. Chandrashekhara, and T.-S. Chua. Atmra: An automatic temporal multi-resolution analysis framework for shot boundary detection. In *MMM*, 2003.

[7] M. B. G. Schindler, L. Zitnick. Internet video category recognition. In *IEEE Workshop on Internet Vision*, 2008.

[8] W. Jiang, C. Cotton, S.-F. Chang, D. Ellis, and A. Loui. Short-term audio-visual atoms for generic video concept classification. In *MM '09: Proceedings of the seventeen ACM international conference on Multimedia*, 2009.

[9] Y. Jun and H. Alex. Exploring temporal consistency for video retrieval and analysis. In *MIR*, 2006.

[10] L. Kennedy. Revision of LSCOM Event/Activity Annotations, DTO Challenge Workshop on Large Scale Concept Ontology for Multimedia. Technical report, Columbia University, December 2006.

[11] Y. Linjun, Y. Yichen, and H. Xian-Sheng. Smart video player. In *IEEE Conference on Multimedia and Expo (ICME),*, 2008.

[12] J. Liu, J. Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". In *IEEE International Conference on Computer Vision and Pattern Recognition*, 2009.

[13] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 2004.

[14] C. D. Manning, P. Raghavan, and H. Schtze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[15] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. In *NIPS '97: Proceedings of the 1997 conference on Advances in neural information processing systems 10*, 1998.

[16] W. Meng, Y. Kuiyuan, H. Xiansheng, and Z. Hong-Jiang. Towards a relevant and diverse search of social images. *IEEE Transactions on Multimedia*, 12(8):829–842, 2010.

[17] W. Meng and H. Xian-Sheng. Active learning in multimedia annotation and retrieval: A survey. *ACM Transactions on Intelligent Systems and Technology*.

[18] W. Meng, H. Xian-Sheng, H. Richang, T. Jinhui, and S. Yan. Unified video annotation via multi-graph learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(5):733–746, 2009.

[19] W. Meng, H. XianSsheng, T. Jinhui, and H. Richang. Beyond distance measurement: Constructing neighborhood similarity for video annotation. *IEEE Transactions on Multimedia*, 11(3):465–476, 2009.

[20] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006.

[21] Z. Qi and S. A. Goldman. Em-dd: An improved multiple-instance learning technique. In *In Advances in Neural Information Processing Systems*, pages 1073–1080. MIT Press, 2001.

[22] R. Rahmani, S. A. Goldman, H. Zhang, J. Krettek, and J. E. Fritts. Localized content based image retrieval. In *MIR '05: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, 2005.

[23] S. Ray and M. Craven. Supervised versus multiple instance learning: An empirical comparison. In *Proceedings of 22nd International Conference on Machine Learning (ICML)*, pages 697–704. ACM Press, 2005.

[24] B. Settles, M. Craven, and S. Ray. Multiple-instance active learning. In *In Advances in Neural Information Processing Systems (NIPS)*, pages 1289–1296. MIT Press, 2008.

[25] S. Siersdorfer, J. San Pedro, and M. Sanderson. Automatic video tagging using content redundancy. In *SIGIR '09: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, 2009.

[26] B. Sigurbjörnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 327–336, 2008.

[27] A. Ulges, C. Schulze, D. Keysers, and T. Breuel. Identifying relevant frames in weakly labeled videos for training concept detectors. In *Proceedings of the 2008 international conference on Content-based image and video retrieval*, CIVR '08, 2008.

[28] A. Ulges, C. Schulze, M. Koch, and T. M. Breuel. Learning automatic concept detectors from online video. *Comput. Vis. Image Underst.*, 114(4):429–438, 2010.

[29] W. Xiao, Z. Wan-Lei, and N. Chong-Wah. Towards google challenge: combining contextual and social information for web video categorization. In *MM '09: Proceedings of the seventeen ACM international conference on Multimedia*, 2009.

[30] L. Yang, J. Liu, X. Yang, and X.-S. Hua. Multi-modality web video categorization. In *MIR '07: Proceedings of the international workshop on Workshop on multimedia information retrieval*, pages 265–274, 2007.

[31] Z.-J. Zha, X.-S. Hua, T. Mei, J. Wang, G.-J. Qi, and Z. Wang. Joint multi-label multi-instance learning for image classification. In *Proceedings of the IEEE international conference on Computer Vision and Pattern Recognition*, pages 01–08, 2008.

[32] Z.-J. Zha, T. Mei, J. Wang, Z. Wang, and X.-S. Hua. Graph-based semi-supervised learning with multiple labels. *Journal of Visual Communication and Image Representation*, 20:97–103, 2009.

[33] Z.-J. Zha, L. Yang, T. Mei, M. Wang, and Z. Wang. Visual query suggestion. In *Proceedings of the ACM international conference on Multimedia*, pages 15–24, 2009.

APPENDIX: the gradient of object function towards the parameter pair

Using the chain rule, each partial derivative can be expressed as:

$$\frac{\delta E_t(W,b)}{\delta w_n} = \sum_{i=1}^{N} \beta[(V_i - Y_i) \sum_{i=1}^{N} \frac{\delta V_i}{\delta S_{ij}} \frac{\delta S_{ij}}{\delta w_n}] \tag{6}$$

$$\frac{\delta E_t(W,b)}{\delta b} = \sum_{i=1}^{N} \beta[(V_i - Y_i) \sum_{i=1}^{N} \frac{\delta V_i}{\delta S_{ij}} \frac{\delta S_{ij}}{\delta b}] \tag{7}$$

The derivative of the video V_i to shot S_{ij} is calculated as follow:

$$\frac{\delta V_i}{\delta S_{ij}} = \frac{(1 + \alpha S_{ij} - \alpha V_i)\exp(\alpha V_i)}{\sum_{j=1}^{l} \exp(\alpha S_{ij})} \tag{8}$$

The final term is the derivative of the shot-level logistic function:

$$\frac{\delta S_{ij}}{\delta w_n} = \alpha S_{ij}(1 - S_{ij}) f_{ij,n} \tag{9}$$

$$\frac{\delta S_{ij}}{\delta b} = V_{ij}(1 - S_{ij}) \tag{10}$$