Mental Visual Browsing

Jun He, Xindi Shang(✉), Hanwang Zhang, and Tat-Seng Chua

School of Computing, National University of Singapore, Singapore, Singapore
{junhe,xindi.shang,hanwang,chuats}@comp.nus.edu.sg

Abstract. We present a surprisingly easy-to-use video browser for helping users to pinpoint a specific video shot in mind, within a long video. At each interactive iteration, the only user effort required is to click 1 shot, which most visually relates to the user's mental target, out of 8 displayed shots. Then, the system updates the browsing model and display another 8 shots for the next iteration. The proposed system is underpinned by a theoretically-sound Bayesian framework that maintains the probabilities of all the video shots segmented from the long video. This framework guarantees that we can find the target shot out of around 1-h video within 3–5 iterations. We believe that our system will perform well in the Video Broswer Showdown game of MMM 2016.

Keywords: Relevance feedback · Bayesian system · Video browsing · Mental search

1 Introduction

Due to the recent advances in visual concept detection [5], we have witnessed a great progress in video search and classification [6]. In these tasks, we generally cast videos into a set of image frames and then consider the whole video as the smallest unit to be operated. However, we neglect that there are also demands for browsing a single long video. This application scenario is typically common in many video editing tasks (*e.g.*, movie or TV post-production). Video Browser Showdown in joint with MMM conference is an interesting venue for researchers to evaluate their systems for video browsing. Video browsing is the interactive process of exploring video content in order to find particular segments [8]. Users are first shown a short video clip (*e.g*, 20 s) and then pinpoint the clip in a long video (*e.g.*, 1–2 h). Most previous browsing tools are designed with sophisticated concept selection [7]. Although these designs are effective for expert users, they are not easy-to-use for novice users and are not flexible for users' nuanced search intentions.

In this paper, we present an easy-to-use video browsing system, called Mental Visual Browsing, which only require user interaction in the way of "click" on the most visually similar shots in 8 displayed shots. This browsing session usually terminates in 3–5 rounds when users find the exact match to their imaginary queries (see Fig. 1). That is to say, users are only required to quickly scan 24–40 images to pinpoint the target shot in a long video containing around 1000–3000

© Springer International Publishing Switzerland 2016
Q. Tian et al. (Eds.): MMM 2016, Part II, LNCS 9517, pp. 424–428, 2016.
DOI: 10.1007/978-3-319-27674-8_44

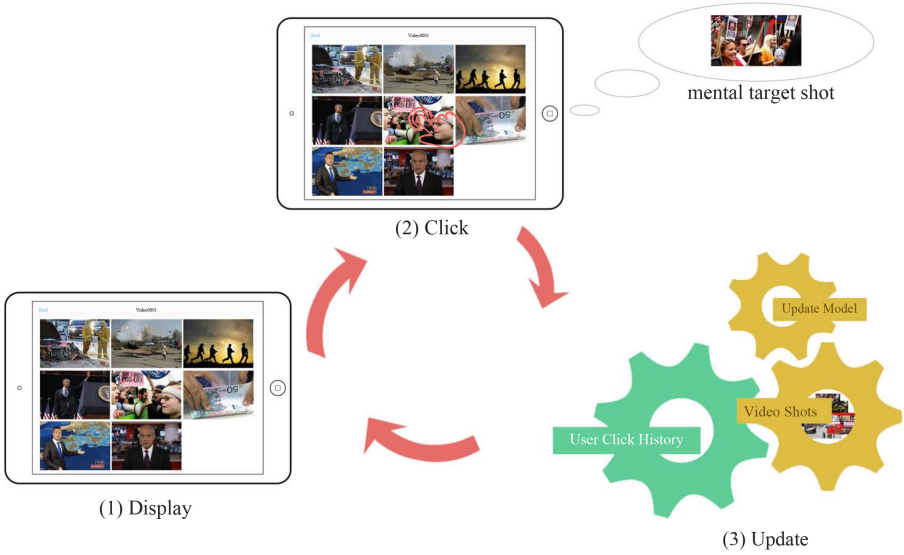

Fig. 1. Our interactive system contains three key parts: (1) Display. 8 shots are displayed to the user; (2) Click. The user click one shot that most relates to the mental target shot; and (3) Update. The underlying algorithm updates the model for the next display. Note that the user effort is considerably limited.

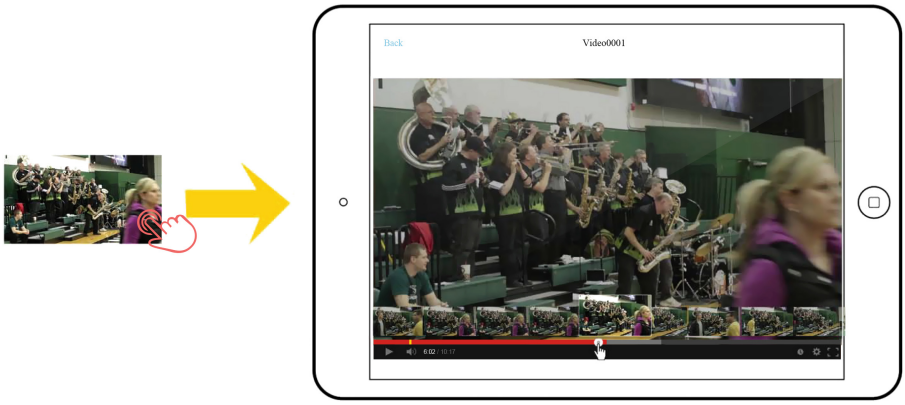

Fig. 2. When the user finds the target shot, the shot can be considered as a starting point for precise localization in the long video.

shots. The interaction is illustrated in Fig. 1. Suppose the long video is segmented into N shots. At the first iteration, the system just randomly select 8 shots to be displayed. After reviewing the 8 shots (by seeing the 8 representative frame image of the shots), the user click the shot that is most visually similar to her target shot in mind. After receiving the click, the system updates the underlying

user intention model and then the model selects the most 8 informative shots (exclusive of the former 8 shots) for the display at next iteration. After the user meets the target shots, she can zoom in to fine-tune the exact locations in the video (see Fig. 2). In summary, we want to highlight that the proposed browsing method is very user-friendly to both experts and novices, and the underlying algorithm can effectively capture users' mental intention through very limited clicks.

2 Video Visual Features

We first segment the long video into shots by using any off-the-shelf shot transition detection softwares. For a typical 1-h video, we obtain 1,000–3,000 shots as our basic units for display. Since our system relies on high-performance visual features for visual representations of users' mental intention, we adopt the recent video features proposed by Xu *et al.* [10]. Specifically, we first extract the frame-level CNN descriptors using Caffe toolkit [4]. The CNN features are using the 7-th fully- connected layer output of VGG-16 networks [9]. Then, we utilize Vector 0f Locally Aggregated Descriptors (VLAD) [3] with $k = 5$ to encode them. We also use Latent Concept Descriptors since it is shown to outperform pure CNN features. We use 256-component k-means centers for VLAD encoding and for all CNN features, we apply PCA-whitening to reduce their dimension into 512. In the same time, signed square root and intro normalization, and L2 normalization are used for post-processing [1]. As a result, each shot is represented by a 256×512-d vector.

3 Statistical Feedback Model

Our goal is to assist users to find their mental target video shot via minimum interactions. With users in the loop, the system will learn how to describe users' intention by modeling the conditional probabilities given users' click on shots. The statistical model presented here is inspired by the one proposed by Ferecatu and Geman [2].

Suppose there are N shots segmented from a long video, denoted as $\mathcal{V} = \{1...i...N\}$ for simplicity. The objective is to identify a shot that matches the semantic and visual impressions in the mind of the user. Let $\mathcal{M} \subset \mathcal{V}$ denote the possible subset of the long video shots, *i.e.*, it can be considered as candidates of users' target mental shots. Of course, $\mathcal{M}$ is unknown or even imaginary to the system. We assume that if a member of $\mathcal{M}$ is displayed, the user will be satisfied and terminates the browsing. At that point, this shot can be used as a starting point for precise localization in the long video. At the t-th iteration, the system displays a shot set $\mathcal{D}_t \subset \mathcal{V}$ to the user. Then, the user can make a feedback to the system by clicking on one of the displayed shots. We denote this feedback as $c_t = k$, where $k \in \mathcal{D}_t$. Moreover, we denote the history displays and user clicks as $\mathcal{H}_t = \mathcal{H}_{t-1} \cup \{c_t = k, k \in \mathcal{D}_t\}$. Our system maintains N-independent

Bayesian system $p_t(i)$ for each video shot i. In particular, $p_t(i)$ is the estimated probability of the event that shot i belongs tho the target set $\mathcal{M}$:

$$p_t(i) = P(i \in \mathcal{M}|\mathcal{H}_t). \tag{1}$$

Next, we detail three key components of our system: (1) **Update**: how to update $p_t(i)$; (2) **Click**: how the system responds to user click $c_t = k$; and (3) **Display**: how to choose shots for displaying $\mathcal{D}_t$.

3.1 Update

By applying Bayes rule, we can rewrite the definition in Eq. (1) as:

$$\begin{aligned} p_t(i) &= P\left(i \in \mathcal{M}|\mathcal{H}_{t-1}, c_t = k, k \in \mathcal{D}_t\right) \\ &= \frac{P\left(c_t = k|\mathcal{H}_{t-1}, i \in \mathcal{M}, k \in \mathcal{D}_t\right) P\left(i \in \mathcal{M}|\mathcal{H}_{t-1}, k \in \mathcal{D}_t\right)}{P\left(c_t = k|\mathcal{H}_{t-1}, k \in \mathcal{D}_t\right)}. \end{aligned} \tag{2}$$

In order to further simplify the above model for computability, we should note several statistical sufficiency and assumptions. First, $c_t = k$ is a sufficient statistic for $k \in \mathcal{D}_t$ since it is obvious that once the user clicks on k, k must be in $\mathcal{D}_t$. Therefore, $P\left(c_t = k|\mathcal{H}_{t-1}, k \in \mathcal{D}_t\right) = P\left(c_t = k|\mathcal{H}_{t-1}\right)$. Second, the fact $i \in \mathcal{M}$ is unrelated to the display set $\mathcal{D}_t$, so we have $P\left(i \in \mathcal{M}|\mathcal{H}_{t-1}, k \in \mathcal{D}_t\right) = P\left(i \in \mathcal{M}|\mathcal{H}_{t-1}\right)$. Third, once given the display $\mathcal{D}_t$, the history $\mathcal{H}_{t-1}$ is no longer informative to the user click $c_t = k$, so we have $P\left(c_t = k|\mathcal{H}_{t-1}, i \in \mathcal{M}, k \in \mathcal{D}_t\right) = P\left(c_t = k|i \in \mathcal{M}, k \in \mathcal{D}_t\right)$ and $P\left(c_t = k|\mathcal{H}_{t-1}, k \in \mathcal{D}_t\right) = P\left(c_t = k|k \in \mathcal{D}_t\right)$. Based on these statistical properties, we have the update model for $p_t(i)$ as:

$$\begin{aligned} p_t(i) &= \frac{P\left(c_t = k|i \in \mathcal{M}, k \in \mathcal{D}_t\right) p_{t-1}(i)}{P\left(c_t = k|k \in \mathcal{D}_t\right)} \\ &= \frac{P\left(c_t = k|i \in \mathcal{M}, k \in \mathcal{D}_t\right) p_{t-1}(i)}{P\left(c_t = k|i \in \mathcal{M}, k \in \mathcal{D}_t\right) p_{t-1}(i) + P\left(c_t = k|i \notin \mathcal{M}, k \in \mathcal{D}_t\right)\left(1 - p_{t-1}(k)\right)}. \end{aligned} \tag{3}$$

Therefore, all we need to update $p_t(i)$ is to model the click $P\left(c_t = k|i \in \mathcal{M}, k \in \mathcal{D}_t\right)$ and $P\left(c_t = k|i \notin \mathcal{M}, k \in \mathcal{D}_t\right)$.

3.2 Click

Let $d(\cdot, \cdot)$ and $s(\cdot, \cdot)$ denote the distance and similarity in the feature space, respectively. Our click probability is modeled as:

$$\begin{cases} P\left(c_t = k|i \in \mathcal{M}, k \in \mathcal{D}_t\right) = \dfrac{s(i,k)}{\sum_{j \in \mathcal{D}_t} s(j,k)}, \\ P\left(c_t = k|i \notin \mathcal{M}, k \in \mathcal{D}_t\right) = \dfrac{d(i,k)}{\sum_{j \in \mathcal{D}_t} d(j,k)}. \end{cases} \tag{4}$$

The intuition behind the above definitions is that if i is the target shot, the probability is enhanced by the similarity between i and click k; if i is not the target, the probability is depressed by the distance between i and k.

3.3 Display

We select the most informative subset $\mathcal{D}_t \subset \mathcal{V}$ by minimizing $\min_{\mathcal{D} \subset \mathcal{V}} \text{Entropy}(z_t|\mathcal{H}_{t-1})$. Intuitively, the selection attempts to make an optimal Voronoi partition for the video shots. The detailed calculation of the probability z_t is given in [2].

References

1. Arandjelovic, R., Zisserman, A.: All about VLAD. In: CVPR (2013)
2. Ferecatu, M., Geman, D.: A statistical framework for image category search from a mental picture. TPAMI **31**(6), 1087–1101 (2009)
3. Jégou, H., Perronnin, F., Douze, M., Sanchez, J., Perez, P., Schmid, C.: Aggregating local image descriptors into compact codes. TPAMI **34**(9), 1704–1716 (2012)
4. Jia, Y.: Caffe: an open source convolutional architecture for fast feature embedding (2013). http://caffe.berkeleyvision.org
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS (2012)
6. Over, P., Awad, G., Michel, M., Fiscus, J., Sanders, G., Shaw, B., Kraaij, W., Smeaton, A.F., Quenot, G.: Trecvid 2012 - an overview of the goals, tasks, data, evaluation mechanisms and metrics. In: TRECVID (2012)
7. Schoeffmann, K.: A user-centric media retrieval competition: the video browser showdown 2012–2014. IEEE MultiMedia **21**, 8–13 (2014)
8. Schoeffmann, K., Ahlström, D., Bailer, W., Cobârzan, C., Hopfgartner, F., McGuinness, K., Gurrin, C., Frisson, C., Le, D.-D., Del Fabro, M., et al.: The video browser showdown: a live evaluation of interactive video search tools. IJMIR **3**(2), 113–127 (2014)
9. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2014). arXiv preprint arXiv:1409.1556
10. Xu, Z., Yang, Y., Hauptmann, A.G.: A discriminative cnn video representation for event detection (2014). arXiv preprint arXiv:1411.4006