Generating video thumbnails, a replacement for video preview images

Master Thesis Information Studies

Jorick van Hees
*
Blue Billywig
jorick.vanhees@student.uva.nl
UvA Student Nr.: 10894020
VU Student Nr.: 2567527

Categories and Subject Descriptors
Computing methodologies [**Computer vision**]: Video thumbnails

1. INTRODUCTION
Thumbnail images for videos are used all over the web. They are small, static representations of videos, and provide a visual preview of the video itself. In combination with a title and description, they form one of the most common interfaces when dealing with a collection of videos. They increase the accuracy when conducting searches in video databases, improve the aesthetics of an overview page, and can increase engagement when using appealing thumbnails.

News media websites often have an overview page (like a front page) that display links to articles containing a video. These links are often presented using the title, a textual description and a thumbnail from the video. This thumbnail is often the only visual reference to the video, and is used to engage the user and improve the 'click-through ratio'.

With the increase of broadband internet speeds and reliability (even on mobile devices), opportunities arise to use short videos as thumbnails. These video thumbnails could increase engagement and user experience compared to static thumbnails.

In order to use video thumbnails in current production workflows as a replacement for static thumbnails, a similar system has to be designed for video thumbnails. In this work, we will propose a way to automatically generate a selection of video thumbnail candidates, based on event detection and concept recognition. The system is then evaluated with a user study, in which the engagement of a video thumbnail is tested against a static thumbnail variant.

Specifically include research question and subquestions?

Do I need to explain this in the introduction?

Manually analysing a video in order to find a proper thumbnail is a time consuming task which an editor doesn't want to be bothered (todo: right word?) with. This is especially true when multiple videos are published each hour. To improve workflow and convenience when selecting a thumbnail, a machine can propose multiple candidates to the editor. This will save time and increases convenience. Such a workflow currently exists for static thumbnails using various algorithms.

1.1 Related work - TODO
Work in the field of video summarisation has a lot of common ground with the generation of video thumbnails. Both use similar data to extract the desired information from a video and its metadata, the same techniques in computer vision is used to process the data and the end result could be very similar.

One could argue that the results from video summarisation could be used in some implementations of the video thumbnails. However, the use cases in both domains are vastly different in terms of user engagement. In general, a summary tries to accurately describe the contents of the video, which eliminates the need to view the full video. In turn, the goal of the thumbnail is to engage the user to view the full video. It tries to show just enough to trigger the user to view the remaining content. The vastly different goal of the video thumbnail has such an impact, that the generation of video thumbnails deserves its own separate task.

1.2 Event detection
Existing systems for video event detection based on the TRECVID Media Event Detection Task often use the provided training data. In our system, no training data is available. However, we can use a number of techniques in order to extract the most characteristic frames from the video.

2. DATASET
(todo)

3. GENERATING VIDEO THUMBNAILS

good vs bad / positive vs negative samples?

The main body of this paper consists of the design of a system that is able to generate video thumbnails, based on metadata, an editor-selected static thumbnails and the video itself. The system can be divided into four major stages: Candidate selection, candidate evaluation and topic-based model training. In the first stage, a number of moments in the video are selected and labeled as candidates. In the second stage, candidates are labeled with a ground truth value (?) based on their similarity to the editor-selected thumbnail. In the third stage, we cluster all videos in the dataset based on topics derived from their metadata. In the fourth and final stage, an SVM is trained for each cluster using the labeled candidates from each video in that cluster.

3.1 Candidate selection

Instead of evaluating all possible frames in a video using a sliding window, a number of candidates are selected from the video. The selection algorithm is inspired by the bag-of-fragments as described in [**Mettes:2015vg**].

More intro

Features from the video are extracted from individual frames with 1 frame per second using a convolutional neural network, trained on the ImageNet dataset [**Krizhevsky:2012wl**] using CaffeNet [**Jia:2014cm**]. The result is a high-level sparse feature representation of the frame with 1000 dimensions. This representation is used throughout the system as a representation of the frame. This method of extracting features proves to be very effective compared to low-level feature extraction for event-detection [**Habibian:2013ks**, **Althoff:2012gf**] and object recognition.

References to object detection.

better explanation of slinding window

A comprehensive list of video thumbnail candidates is generated using a brute-force sliding window method. The video frames in the window are max-pooled into a vector representation to catch every concept in the video thumbnail. In order to prune the list of candidates, the video thumbnail is compared to a bag-of-fragments representation of the full video. The bag-of-fragments are created with the features extracted from all frames, clustered using K-means. The number K is calculated using the length of the video $K = \frac{frames}{20 * fps}$. The similarity between the video thumbnail and each bag-of-fragments is measured using cosine similarity. The resulting similarity vector (with K dimensions) is then normalised, to prioritise the most representing video thumbnail for each bag-of-fragments. Finally, a ranking is made for each bag-of-fragments where the top result in each ranking is selected as thumbnail candidate.

3.2 Candidate evaluation

The resulting list of candidates is ranked based on the human-selected static thumbnail for that specific video. The same method of extracting features from the video is used on the static thumbnail, which results in a concept vector that can be compared to concept vectors extracted from the video frames. For every candidate, the cosine similarity between the candidate vector representation (as described in 3.1) and the thumbnail concept vector is calculated.

Since the resulting similarity values are very diverse across the dataset, a top N % percentile selection is used to determine positive (+1) and negative (-1) values which can later be used to train the SVM models. The N is selected based on the graph in figure (insert figure).

Graph about N percentile in candidate evaluation

3.3 Topic clustering

Since the tags available in the metadata are not discriminative and mainly used for web search engine optimisation, it is not possible to categorise the videos based on these tags. Instead, the videos are clustered using all available metadata. The title, description and tags are concatenated and a stoplist is used to remove regular words. A bag-of-words corpus is formed with the resulting documents, which is used to create a latent Dirichlet allocation model.

With the LDA model, a sparse vector representation of the video metadata is generated which can be used to cluster the videos in the dataset. Clustering is done using K-means, where K is selected based on the graph in figure (insert figure). A larger K would mean more specific topics and more accurate SVM predictions, but will also reduce the number of samples for each topic. We found that the ideal value of K would be around 10, where we wouldn't overfit the data and still gain a reasonable accuracy of the SVM's (insert figure about SVM accuracy versus number of K?).

insert figure about SVM accuracy versus number of K

3.4 Model training

For each topic created in the topic clustering, an SVM is trained in order to rank new video thumbnail candidates. The data used for training an SVM consists of all the videos that are classified in that specific cluster, along with the labels generated in 3.2. In order to avoid overfitting the data, we use the same parameters for each SVM. An RBF kernel is used with $C = 10$ and $\gamma = 100$ on an average dataset of X training samples. The average accuracy of all SVM's was around 0.75.

insert avg number of training samples & check average accuracy

4. USER STUDY

include numbers from study

The video thumbnails generated by the system described in 3 have been tested in an A/B user survey against a baseline in the form of static thumbnails: 50% of the respondents received the survey which included video thumbnails, while the other 50% received a version with static thumbnails.

The survey was conducted via a custom build website to ensure compatibility across multiple devices.

4.1 Survey setup

For each video, two previews were shown in successive order: The first preview contained only a title and description, while the second preview contained a title, description and a (static or video) thumbnail. An example of the latter preview is shown in figure...

include figure with preview

After each preview, two statements were made about the video preview to measure the engagement of the participant towards the video, and wether the participant felt informed about the contents of the video:

- I am interested in viewing the video (engagement).
- I know what to expect from the video (informative).

By comparing the difference in answers between the version with thumbnail and without thumbnail, we are able to measure the impact of using a thumbnail in the preview. This difference can then be compared between the static thumbnail and video thumbnail, allowing us to analyse the effect of a video thumbnail. This way, any preconception from the user about certain topics or videos can be taken into account.

The videos used in this survey were manually picked from the dataset based on number of views. Early feedback on the survey setup revealed that randomly selected videos would be uninteresting, regardless of the form of the video preview. The difference in target audience between the dataset source and survey respondents would be the primary explanation. The age of most of the videos is a second, since most of the news videos are outdated at the point of conducting the survey.

4.2 Responses

A total of (??) respondents participated, of which (??) received the version with video thumbnails, and (??) received the version with static thumbnails. Each participant received previews for a total of three videos, resulting in (??) responses for the static thumbnail, and (??) responses for the video thumbnail.

4.3 Result analysis

5. CONCLUSION

6. FUTURE WORK