Story-based Video Retrieval in TV series using Plot Synopses

Makarand Tapaswi
makarand.tapaswi@kit.edu

Martin Bäuml
baeuml@kit.edu

Rainer Stiefelhagen
rainer.stiefelhagen@kit.edu

https://cvhci.anthropomatik.kit.edu/projects/mma
Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany

ABSTRACT

We present a novel approach to search for plots in the storyline of structured videos such as TV series. To this end, we propose to align natural language descriptions of the videos, such as plot synopses, with the corresponding shots in the video. Guided by subtitles and person identities the alignment problem is formulated as an optimization task over all possible assignments and solved efficiently using dynamic programming. We evaluate our approach on a novel dataset comprising of the complete season 5 of Buffy the Vampire Slayer, and show good alignment performance and the ability to retrieve plots in the storyline.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

Keywords

story-based retrieval; text-video alignment; plot synopsis

1. INTRODUCTION

Searching through large sets of videos is an active field of research and has come a long way from content- to concept-based retrieval [21, 22]. However, most of the research works either on structured videos such as broadcast news [13, 18] or unstructured web videos (*e.g.* as originating from YouTube) [20]. In both scenarios the retrieval is essentially restricted to *concepts* or *events* where a certain low- to mid-level description (*e.g.* "Shots containing map of USA" or "Shots showing George Bush entering a car") is queried.

The above however, is not well suited to search for specific plots or stories (*e.g.* "Gandalf falls to a Balrog of Moria" or "Batman climbs out of the prison pit" or "Obi-Wan cuts Darth Maul in two with his lightsaber") in a continuous storyline that exists in most TV series and movies. In this paper we take a step towards addressing the problem of finding story events in large collections of video. We propose to perform automatic retrieval of events in the storyline by leveraging existing human descriptions on Wikipedia or other fan sites. Note that these descriptions or *plot synopses* (*e.g.* `en.wikipedia.org/wiki/Buffy_vs._Dracula#Plot`) are distinctly different from other textual sources of information such as subtitles or transcripts (*e.g.* `www.buffyworld.com/buffy/transcripts/079_tran.html`) as they are rich in content, and have the complex narrative structure of a story. We will show that the plot synopses are quite effective in capturing the story and help in performing retrieval.

As a precursor to the retrieval task, we need to know which sentence corresponds to which part of the video. We propose to align sentences of a plot synopsis to shots in a video by using elements which occur both, in the visual as well as the textual depictions. Such elements are primarily characters in the story, the location of the event, key objects and actions. We focus our attention towards characters as they are the most important factors in shaping any story.

The alignment also opens up novel ways to approach other existing applications. One such application is *semantic video summarization*. After alignment of the complete video and synopsis, a text summarizer first shortens the plot synopsis. Standard video summarization techniques [15] can now only consider the shots that are aligned to the retained sentences. The major difference and advantage is that the summary is now directly based on the story. The alignment is also an interesting source for automatically generating labels for training data. For example, as in [12], we can train high level action recognition concepts such as *persuade*, *unpack* or *play*; or even attempt to train models to automatically generate high-level descriptions of similar video content.

The main contributions of this paper are:

1. To the best of our knowledge, we present the first approach to align human-written descriptions such as plot synopses to shots in a video *and* to perform automatic retrieval of video snippets of story events in TV series from textual queries.
2. We develop a generic approach to perform the alignment between text sentences and video shots using shared textual and visual cues and propose an efficient solution to estimate the optimal alignment.
3. We contribute a novel data set of human-annotated alignments of plot synopses from the web for the complete 5th season of the TV series *Buffy the Vampire Slayer* (see Sec. 5).

The rest of the paper is organized as follows. We first present some work related to our problem in Sec. 2. The

ICMR '14, April 01 - 04 2014, Glasgow, United Kingdom

ACM 978-1-4503-2782-4/14/04 $15.00.
http://dx.doi.org/10.1145/2578726.2578727

preprocessing steps for both text and video are briefly presented in Sec. 3.1. We describe the cues, namely character identities (Sec. 3.2) and subtitles (Sec. 3.3) to guide the alignment which is performed using techniques presented in Sec. 3.4. Sec. 4 describes how the aligned text and video elements are linked to search for story plots. Finally, we evaluate our approach in Sec. 5 and conclude in Sec. 6.

2. RELATED WORK

Retrieval. Over the years the TRECVid challenge [20] has promoted video retrieval through tasks such as Multimedia Event Detection (MED) or semantic indexing. Already on the path towards better understanding, the focus of the retrieval has shifted to concepts [21] rather than low-level content features.

More recently, there is a shift towards leveraging crowdsourcing to improve image/video search. In the domain of concert videos, Freiburg *et al*. [8] facilitate easy navigation within videos by augmenting automatically detected concert concepts with a user-feedback system. Wang *et al*. [25] propose a method to learn a joint latent space of image-text pairs for topic models which are crowd-sourced from Wikipedia.

Aligning text to video. In previous work on TV series and movies, transcripts have been used as additional textual source of information. However, aligning transcripts to videos is a relatively simple task and can be achieved by using subtitles as an intermediary, since both transcripts and subtitles essentially contain the same dialogs [4, 5]. Transcript alignments are used successfully in tasks such as person identification [3, 5] or action recognition [12]. Recently Liang *et al*. [16] used transcripts to index a video database with characters, place and time information. Given a new script they use this data followed by post-production to automatically generate new videos.

In the domain of sports videos, webcast text has been used for event detection [26]. This is a relatively easy task since the webcast includes a time tag in the descriptions and the video typically overlays time information.

In our case, note that TV series transcripts are very different from plot synopses and do not outline the story, but describe the setting, characters and dialogs. On the other hand, plot synopses contain semantically rich descriptions which offer opportunities to better understand the video content, at the cost of harder alignment methods.

Automatic description of images and videos. There has been some work to generate video descriptions for specific domains of video, although the descriptions are quite simple. For example, Gupta *et al*. [10] propose a technique to describe sports videos by combining action recognition and modeling the events as AND-OR graphs. Tan *et al*. [23] use audio-visual concept classifiers for a few hand-picked set of concepts, followed by rule-based methods to generate descriptions. On a related problem of describing images, Farhadi *et al*. [6] use an intermediate image representation which contains triplets of key objects, actions and scenes in the image. Habibian *et al*. [11] recently demonstrate a method to convert videos to sentences and vice-versa. They use a large (1346) number of concepts which act as intermediaries between the text and video.

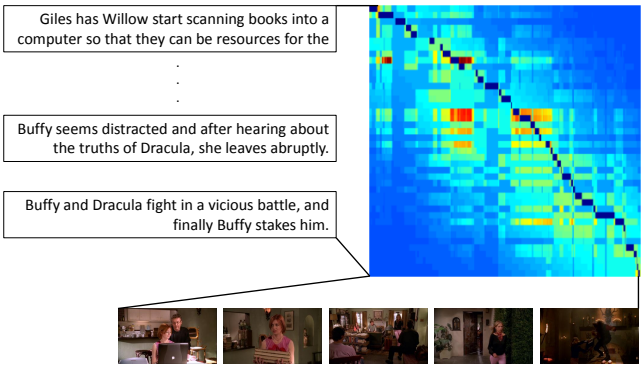

Figure 1: Sentences (rows) are aligned to shots (columns). The figure shows a sample similarity matrix $f(\cdot,\cdot)$ overlaid with the annotation.

In this paper, we employ character identities as our "concepts" and use the structure in TV series to align shots to sentences. We use subtitles as a second set of cues to improve the alignment performance. The retrieval is performed by matching the query to the text and fetching the corresponding video parts.

3. TEXT-VIDEO ALIGNMENT

We now describe our approach towards guided alignment of shots in the video to sentences in the plot synopsis. This alignment is the groundwork of the story-based retrieval. The goal is to determine for each sentence s_i of the plot synopsis, the set of corresponding shots $T_i = \{t_{i1}, \ldots, t_{ik}\}$ that depict the part of the story that the sentence describes. Using subtitles and character identity cues as a common intermediate representation, we formulate a similarity function $f(s_i, t_j)$ between a sentence s_i and a shot t_j.

Fig. 1 presents an overview of the alignment problem, where sentences in the synopsis are stacked as rows, and shots from the video as columns. We also see the similarity matrix between shots and sentences overlaid with the ground truth shot to sentence assignment.

3.1 Pre-processing

Plot Synopsis. We can obtain plot synopses for most TV series episodes or movies from Wikipedia articles or other fan-sites such as movie/series specific `wikia.com`. A list of characters in each episode is also collected from sources such as IMDb and Wikipedia. This aids in improving the person identification results and is necessary for the alignment. We perform part-of-speech tagging on the plot synopses using Stanford CoreNLP [1] toolbox. The tags are used to determine proper nouns (NNP) and pronouns (PRP) to obtain a list of names in the text.

Video. While a sentence forms the basic unit of processing for the plot synopsis, a shot is used as the basic unit for the video. The video is first divided into shots, *i.e*. a set of continuous frames between two cuts, using a normalized version of the *Displaced Frame Difference* [27]

$$DFD(t) = \|F(x,y,t) - F((x,y) + D(x,y), t-1)\|, \quad (1)$$

where $D(x,y)$ is the optical flow between frames $F(x,y,t-1)$ and $F(x,y,t)$, (x,y) the pixel and t time. The DFD represents the difference between consecutive motion compensated frames. We filter the DFD and threshold local maxima to obtain shot boundaries.

From the video data, we also extract the subtitles which are available on most DVDs. We collect transcripts from fan websites, containing minimal information – who is speaking what – to perform unsupervised person identification [3].

3.2 Character Identification

Characters and their interactions play a major role in the depiction of any story. This is also reflected in recent video summarization techniques [19] which are turning towards identifying characters prior to summarization.

Similarly, character identities are also most influential in the alignment between sentences in the plot synopsis to shots in the video. A reference to a character in a sentence of a plot synopsis indicates a high likelihood of him/her appearing in the corresponding set of shots. We observe that in the generic structure of sentences – subject, verb and object – most sentences in the plot contain a reference to the characters either as the subject or object, or even both.

Extracting identities from text. To resolve pronouns and other character references (*e.g.* sister, father, etc.), we perform co-reference resolution [14] and cluster the nouns attaching them with a name. This is augmented by a simple, yet surprisingly effective technique of looking back for the antecedent that agrees in gender. For example, in the following sample from a plot synopsis

"*Buffy awakens to find Dracula in her bedroom. She is helpless against his powers and unable to stop...*"

we see that *She* and *her* refers to *Buffy* and *his* to *Dracula*.

We compare this technique of obtaining a list of names in each sentence against human annotations – a list of names inferred by reading the plot only. We observe that on average across the episodes, a plot synopsis consists of ∼80 names. Our method is able to detect them with a recall of 73% and a precision of 82%. A couple unresolved problems with this simple approach are (i) plural pronoun references such as *they* which are not resolved; and (ii) references to people who are being talked about, but are not physically present, *e.g.* "*Riley asks Spike about Dracula ...*". Here *Dracula* does not appear in the scene. To the best of our knowledge, there are no NLP approaches that are able to perform (i) plural pronoun resolution and/or (ii) semantic interpretation, and handle these types of errors.

Extracting identities from video. We perform face tracking with a particle filter [3] using frontal, half-profile and profile MCT-based face detectors [9]. Following the principle from [3], we first align subtitles (what is spoken when) and transcripts (who speaks what) to obtain who speaks when, and subsequently tag face tracks in an unsupervised manner with identities based on lip motion analysis. For recognition of all face tracks, we build on a local appearance-based approach and train SVM models [24] for each character. We thus obtain character identities in a fully-automatic setting without any supervision.

Each episode, typically ∼40 minutes in duration, contains on average ∼950 face tracks and we recognize them correctly (among a set of 59 characters) with an accuracy of 63%.

Similarity function. The number of times characters appear in a video varies widely. Primary characters appear throughout the video and are referenced often in the text, thus making them weak sources of information to pin-point the precise alignment between shots to sentences. On the other hand, characters from the secondary cast appear for a limited duration in the video, and are rarely referenced in the text. Therefore, when they appear, they provide strong cues towards the alignment.

We thus rate the importance of each character c^* as

$$\mathcal{I}(c^*) = \frac{\log(\max_c n_{FT}(c))}{\log(n_{FT}(c^*))}, \tag{2}$$

where $n_{FT}(c)$ is the number of tracks assigned to c.

Further, owing to the standard editing practices used in TV series and movies, in many shots we do not see all people that are actually in the scene. For example, in shot sequences with long conversations it is common to see the speakers in turns (and see "over the shoulder" of the person who is currently not speaking). To determine the presence of characters in such settings, it is helpful to also consider the neighborhood of a shot. Thus, if character c appears in shot j, we spread his/her influence to a few neighboring shots $j-r, \ldots, j, \ldots, j+r$. We empirically choose $r = 4$.

The similarity function for identities between each sentence s_i and shot t_j is given by

$$f_{id}(s_i, t_j) = \sum_{k=j-r}^{j+r} \sum_{c\in\mathcal{C}} \sum_{d\in\mathcal{D}} \delta_{cd} \cdot \mathcal{I}(c), \tag{3}$$

where $\mathcal{C}$ is the set of characters seen in the r neighborhood of shot j and $\mathcal{D}$ is the list of names obtained from sentence i. δ_{cd} is the Kronecker delta and is 1 iff $c = d$.

3.3 Subtitles

We use *subtitles* as another cue to align the shots in the video to the plot synopsis. While plot synopses are natural language descriptions of the story, subtitles provide information about the time and dialog between characters. We observe that the plot synopses rarely contain direct references to the dialog, nevertheless, keywords such as names, places, or object references help to perform the alignment.

We first normalize the two texts (subtitles and plot synopses) such that the basic unit for matching is a word and perform stop-word removal. We then assign subtitles to shots based on their timestamps. A similarity function (like the one for character identities f_{id}) is computed between every sentence i from the plot synopses and shot j, by counting the number of matches between words v in sentence i and w in the subtitles that are assigned to shot j as

$$f_{subtt}(s_i, t_j) = \sum_{v\in s_i} \sum_{w\in subtt\in t_j} \delta_{vw} \quad . \tag{4}$$

3.4 Alignment

Given the similarity functions between sentences and shots, we turn to the problem of finding shots corresponding to the sentences. We define the task as an optimization problem over all possible assignments $\mathcal{M} \in (\mathcal{S} \times \mathcal{T})$ between the set of sentences $\mathcal{S}$ and the set of all possible combinations of shots $\mathcal{T}$. We are interested in finding the assignment $\mathcal{M}^*$ that maximizes the joint similarity $\mathcal{J}$ between assigned sentences and shots

$$\mathcal{M}^* = \operatorname*{argmax}_{\mathcal{M}} \mathcal{J}(\mathcal{M}) \tag{5}$$

$$= \operatorname*{argmax}_{\mathcal{M}} \left[g(\mathcal{M}) \cdot \sum_{(\mathcal{S},\mathcal{T})\in\mathcal{M}} f(\mathcal{S},\mathcal{T}) P(\mathcal{S},\mathcal{T}) \right] \quad , \tag{6}$$

where $g(\cdot)$ is a general function that restricts possible assignments on a global level, *e.g.* to discourage assigning every shot to every sentence. $P(\cdot,\cdot)$ serves as a prior for individual assignment pairs. For example, it is highly unlikely that the first sentence describes the last 5 shots of the video, *i.e.*, the prior for such an assignment is usually low.

Note that without $g(\cdot)$, if the similarity functions are strictly positive (as the ones we define in the previous section), $\mathcal{M}^*$ will always result in the assignment that connects all shots to all sentences, an obviously futile solution.

In general the number of sentences N_S is much smaller than the number of shots N_T. Thus, it is certainly possible that multiple shots belong to the same sentence, while it is uncommon that one shot is described by multiple sentences (this depends of course on the style of the plot synopses). We therefore discourage such assignments by choosing

$$g(\mathcal{M}) = \begin{cases} 1 & |\mathcal{S}| \leq 1 \;\forall\; (\mathcal{S},\mathcal{T}) \in \mathcal{M} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

which we use through the following unless otherwise noted.

We now discuss four alternatives for performing the alignment. We compare against one baseline – a temporal prior and propose two efficient strategies for optimizing $\mathcal{J}(\mathcal{M})$ given g, f and P.

Temporal Prior. The simplest strategy to perform the alignment is to equally distribute the shots to sentences. We evaluate this as our baseline. Note that we do not need any shared cues in this method. Let N_T be the total number of shots in the video and N_S the total number of sentences in the plot synopsis. We then assign $n = N_T/N_S$ shots to each sentence (assuming $N_S < N_T$). Thus, shot t_j is assigned to sentence s_i with

$$i = \lceil j/n \rceil \ . \tag{8}$$

This assignment is equivalent to setting $f(\cdot,\cdot) := 1$, and using a normally distributed temporal prior

$$P(s_i, t_j) \propto \exp\left[-\frac{(j-\mu_i)^2}{2\sigma^2}\right] \tag{9}$$

with $\mu_i = (i - \frac{1}{2}) \cdot n$. We empirically select $\sigma = n$ and keep g as in Eq. 7 to allow only one sentence to be assigned to a shot. Fig. 2 (PRIOR) shows an example of the resulting assignment obtained on one of the episodes in our dataset. Note how the assignment is restricted to the diagonal.

We use Eq. 9 as prior in the joint similarity (Eq. 6) for all the following methods in order to discourage unlikely assignments.

Max Similarity. Max Similarity maximizes the joint similarity. We assign each shot to exactly one sentence (see Eq. 7) such that the sum over the similarity functions of the assigned sentences and shots is maximized.

Fig. 2 (MAX) shows the result of Max-Similarity. Note how the alignment "jumps" between sentences since we do not constrain the temporal order of the assignment. Also note how the alignment is nevertheless constrained to the leading diagonal by the prior (Eq. 9).

Max Similarity with Temporal Consistency. A problem with the Max Similarity alignment technique is that it treats shots independently. Clearly, that is not true, as it is very likely that two consecutive shots belong to the same

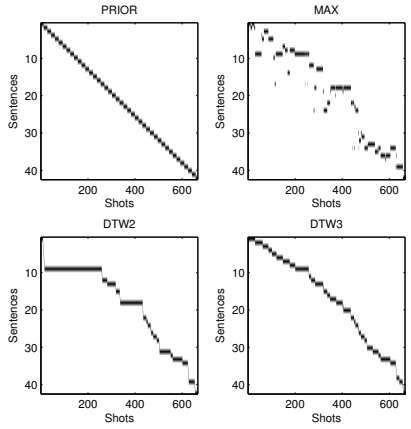

Figure 2: Top: PRIOR and MAX; bottom: DTW2 and DTW3 alignments for BF-01.

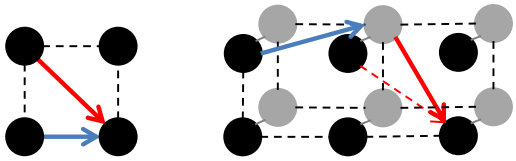

Figure 3: LEFT: Paths that our simple DTW can take (blue, light) sentence continuation, (red, dark) start new sentence. RIGHT: Paths that DTW3 can take. We represent the second layer in gray. Sentence continuation now also changes layer, and new sentences always start at the topmost layer.

sentence. In order to make the alignment temporally consistent, we allow to assign a shot t_{j+1} only to the same sentence s_i as t_j, or to the next sentence s_{i+1}.

$$g(\mathcal{M}) = \begin{cases} 1 & |\mathcal{S}| \leq 1 \;\forall\; (\mathcal{S},\mathcal{T}) \in \mathcal{M} \text{ and} \\ & i \leq m \leq (i+1) \;\forall\; (s_i,t_j),(s_m,t_{j+1}) \in \mathcal{M} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

In order to optimize this efficiently we use dynamic programming, specifically a modified version of the DTW algorithm [17].

DTW2. Consider the similarity function $f(\cdot,\cdot)$ as a matrix of size $N_S \times N_T$, for N_S sentences and N_T shots. Each point of this matrix represents the similarity of one shot to one sentence.

Temporal consistency (Eq. 10) is enforced by allowing only two paths to arrive at every point on the DTW grid. The first option is to assign the current shot to the same sentence as the previous shot, while the alternative is to assign it to a new sentence. This is illustrated in Fig. 3 (left).

We construct a matrix D based on the similarity function f following the above movement rules:

$$D(i,j) = \max \begin{cases} D(i,j-1) + f(s_i,t_j) \\ D(i-1,j-1) + f(s_i,t_j) \end{cases} \tag{11}$$

The highest scoring path on this matrix D induces the optimal assignment $\mathcal{M}^*$ of the shots to sentences which can be found by backtracking. The computational complexity

of this algorithm is in $\mathcal{O}(N_S N_T)$. Fig. 2 (DTW2) shows the resulting alignment.

DTW3. While DTW2 does find temporally consistent assignments, it does not restrain the number of consecutive shots assigned to one sentence. As we see in Fig. 2 (DTW2), this can lead to highly irregular assignments.

Consider a sentence s_i which contains many instances of names from the cast. In such a case, it is likely that $f(s_i, \cdot)$ has a high value of similarity throughout all shots. Therefore, the method has a tendency to assign a large number of shots to this sentence. However, in reality it is unlikely that one sentence describes a large proportion of shots.

To prevent this, we make a modification to the DTW2 algorithm. We introduce a decay factor α_k which depends on the number of shots already assigned to that sentence. The maximum number of shots to which a sentence can be assigned is set to $z = 5n$ (5 times the expected average shot length of a sentence $n = N_T/N_S$), and the weights are computed as follows

$$\alpha_k = 1 - \left(\frac{k-1}{z}\right)^2, \qquad k = 1, \ldots, z \tag{12}$$

This can still be formulated as a dynamic programming problem, and thus be solved efficiently. To incorporate the decay in D, we extend the matrix to a third dimension $k = 1, \ldots, z$. The valid transitions (see Fig. 3 (right)) are

$$D(i, j, k) = D(i, j-1, k-1) + \alpha_k f(s_i, t_j), \quad k > 1 \tag{13}$$

to assign a shot to the same sentence and

$$D(i, j, 1) = \max_{k=1,\ldots,z} D(i-1, j-1, k) + f(s_i, t_j) \tag{14}$$

to assign a shot to a new sentence. Note that the first shot of every sentence is forced to start at $k = 1$. We compute the forward matrix D and then backtrack to obtain the best alignment. The computational complexity of DTW3 is in $\mathcal{O}(N_S N_T z)$[1]. Fig. 2 (DTW3) shows an example of the resulting assignment. Note the difference to DTW2 where we have one sentence being assigned a large number of shots.

4. STORY-BASED SEARCH

Based on the alignment between shots and sentences, story-based video search can be reduced to a text search in the plot synopses. We use Whoosh 2.5.4 [2], a full text indexing and search library.

We first index the sentences of the plot synopses as independent documents. We also index groups of 2 sentences and 3 sentences taken at a time as documents to facilitate queries which can span a larger time. The scoring is performed using the BM25F [28] algorithm. We define the search to return scores and highlight matched terms for convenience. Clicking on the any of the returned documents (sets of sentences) brings up the corresponding set of aligned shots from the video.

Evaluation measures. We evaluate our story-based search from the point-of-view of a user and measure:
(i) **top 5**: the ability of the user to find the queried story event in the top 5 returned documents (as a 0-1 answer) and

[1]For $z \sim 100$, $N_S \sim 40$ and $N_T \sim 700$ DTW3 takes a couple of minutes to solve with our (unoptimized) Matlab implementation.

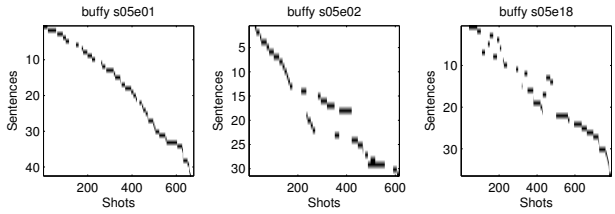

Figure 4: Ground truth annotations for BF-01, BF-02 and BF-18.

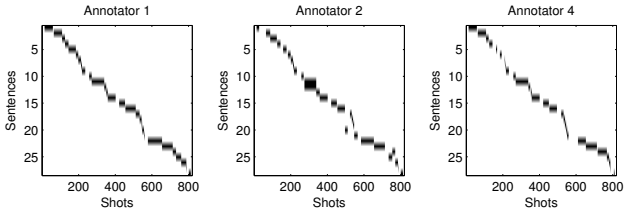

Figure 5: Variation in ground truth across annotators for BF-03. The Fleiss κ inter-rater agreement is 0.701 indicating substantial agreement.

(ii) **time deviation**: the difference in time between the set of returned shots from the ground truth position of the story event (smaller the better). In case of overlap, we indicate the percentage of time which is common for both.

5. EVALUATION

5.1 Experimental Setup

Dataset We evaluate the proposed alignment between plot synopses and shots of a video and the story-based retrieval on the complete season 5 of the TV series *Buffy the Vampire Slayer* (BF). Each episode of the series is a self-contained story, which contributes towards a larger storyline lasting for one season. The season contains 22 episodes which are about ∼42 minutes each. The average number of sentences in the plot synopses are 36 and vary from 22 to 54 in some episodes. We will make the dataset publicly available.

Alignment performance criterion. We propose assignment accuracy as a simple criteria to measure the performance of an alignment method. Let $\mathcal{A}(t_j)$ represent the predicted sentence $\hat{s}$ to which shot t_j is assigned and $\mathcal{G}(t_j)$ the ground truth sentence s. The measure counts the number of shots assigned to the correct sentence divided by the total number of shots.

$$ACC = \frac{1}{N_T} \sum_j \delta(\mathcal{A}(t_j), \mathcal{G}(t_j)) \tag{15}$$

We use alignment accuracy ACC (higher is better) to compare the different alignment methods from Sec. 3.4. All methods are also compared against human-annotated alignments.

Ground truth annotations. The assignment of video shots to plot synopses sentences is inherently subjective, and different people have varied opinions about the set of shots which correspond to a sentence. We study the effect of the subjective nature of the problem by obtaining alignment labels – mapping of shots to sentences – from four different annotators on the first 4 episodes, BF-01 to BF-04. We

Table 1: Comparison of alignment accuracy against various alignment techniques.

Method		BF-01	BF-02	BF-03	BF-04
Human		81.49	86.36	77.52	72.81
Prior		2.95	23.78	27.90	8.82
Character ID	MAX	11.58	30.89	23.60	19.12
Character ID	DTW2	9.40	35.02	18.75	28.43
Character ID	DTW3	**42.22**	43.75	**40.43**	**40.30**
Subtitles	DTW3	20.43	**48.42**	35.30	30.08
Char-ID+Subt.	DTW3	40.82	**51.26**	**41.43**	**47.58**

evaluate our proposed methods against all annotators and present averaged results. We also collect alignment labels from one annotator for the entire season, BF-01 to BF-22.

Fig. 4 shows ground truth annotations by one annotator for BF-01, BF-02 and BF-18. The graphs show the alignment path, *i.e.* the assignment of shots (on x-axis) to sentences (on y-axis). From the annotations, we derive some interesting observations:

1. not all shots need to be assigned to a sentence (seen from the gaps in shot assignment)
2. the video and description need not follow sequentially, there can be jumps between them (BF-02, BF-18)
3. although rare, multiple sentences can be used to describe the same set of shots (BF-02, sentence 28-29)

While the above contradict with some of our assumptions made in Eq. 10, the number of times these violations occur is few. Note however, that our more general formulation of the problem (Eq. 6) allows for all of the above. The above problems are out of scope of the current work.

Cross annotator variation. In Fig. 5 we study the differences in the alignment labels obtained from three different annotators for the same episode BF-03. Note that while the overall structure looks similar, there are differences among them as a few shots are assigned to different sentences.

We compute the Fleiss κ [7] inter-rater agreement for all the videos by considering each shot as a *sample* and the assigned sentence as the *category*. We also use an additional *null* category to include shots that are not assigned to any sentence. The κ values are 0.80, 0.83, 0.70, and 0.70 BF-01 to BF-04 respectively indicating substantial agreement (0.61-0.80) to almost perfect agreement (0.81-1.00).

We also evaluate the difference between annotators as a "Human" alignment performance (see Table 1). The scores are obtained by comparing all annotators against each other in pairs, and then averaging the results. We can consider the "Human" score as an upper bound for achievable performance.

Queries for story-based retrieval. We collect a total of 62 queries related to story events in the complete season. A portion of the queries are obtained from a fan-forum based on the TV series *Buffy* (http://www.buffy-boards.com), while the others are contributed by the annotators of the alignment. The annotators were instructed to only look at the video and not the plot synopsis, while creating additional queries.

The ground truth information for the queries includes the episode number and the time duration in minutes during which the plot unravels.

Table 2: Alignment accuracy on all episodes.

Episode	Prior	Subtitles DTW3	Character ID DTW3	Char-ID+Subt. DTW3
BF-01	2.80	21.39	**40.85**	**42.77**
BF-02	20.29	**41.88**	39.12	**48.05**
BF-03	27.93	31.71	**32.32**	**32.68**
BF-04	4.20	24.37	**37.81**	**42.16**
BF-05	4.30	39.85	**45.33**	**51.11**
BF-06	7.65	33.02	**34.36**	**35.17**
BF-07	12.37	**52.15**	31.06	**55.43**
BF-08	12.73	**39.67**	36.69	**42.98**
BF-09	4.67	40.21	**40.96**	**48.80**
BF-10	5.71	**45.35**	43.23	**50.73**
BF-11	4.26	**50.73**	45.14	49.80
BF-12	9.54	41.91	**45.67**	**55.96**
BF-13	5.69	37.29	**48.67**	**61.62**
BF-14	1.89	**46.14**	21.27	**51.97**
BF-15	20.29	45.89	**57.56**	**60.34**
BF-16	9.66	27.70	**43.31**	**49.63**
BF-17	12.76	57.34	**64.69**	**69.93**
BF-18	6.06	27.27	**38.13**	**39.77**
BF-19	16.35	32.97	**54.59**	**62.16**
BF-20	10.00	19.79	**38.94**	**39.79**
BF-21	2.54	13.94	**34.51**	**51.83**
BF-22	20.75	**43.38**	31.54	38.92
Average	10.11	37.00	**41.17**	**49.16**

5.2 Alignment Performance

Fig. 2 shows a sample alignment result from each of the three different methods. The PRIOR method (top-left) assumes that video shots and sentences are sequential and equally distributed, while the MAX (top-right) picks the best local assignment of shots to sentences. However MAX does not consider temporal consistency, and thus appears fragmented. DTW2 (bottom-left) and DTW3 (bottom-right), unlike MAX, link neighboring shots. In addition, DTW3 constrains the number of shots assigned to each sentence.

Alignment methods. We evaluate various methods discussed in Sec. 3.4 in combination with the two different cues – subtitles and character identities – and present the results in Table 1. The results are obtained by averaging the alignment accuracy across the four different annotators.

We observe that the *Prior* can sometimes perform quite well (specially BF-02, BF-03), when shots and sentences are equally distributed in the plot synopsis. When using automatic character identification (*Character ID*) as the cue for alignment, we see that DTW3 consistently outperforms other methods. In general, *Subtitles* seem to perform worse than *Character ID* based alignment. However, a simple fusion scheme *Character ID + Subtitles* which weights and combines the two similarity functions

$$f_{fus}(s_i, t_j) = f_{id}(s_i, t_j) + 2 \cdot f_{subtt}(s_i, t_j) \qquad (16)$$

performs best on average.

Accuracy for the complete season. Table 2 shows alignment accuracy for all episodes in the season. The evaluation is performed by comparing to only one annotator for all episodes. We see again that *Character ID* outperforms *Subtitles* in most episodes (15 of 22). Note that the fusion of the two cues tends to produce the best alignment accuracy (20 of 22 episodes) irrespective of which method performed better. With a relaxed metric, where we allow alignment within ± 1 sentence, we observe an average accuracy of 71% vs. 49%.

Table 3: Story-based retrieval performance on sample queries from season 5 of Buffy the Vampire Slayer. E01:m35-36 means minutes 35-36 of episode 1. (33) indicates sentence number 33.

#	Query	Ground Truth		Retrieval		Time
		Location	Sentence	top 5	Sentence	deviation
1	Buffy fights Dracula	E01:m35-36	(33) Buffy and Dracula fight in a vicious battle.	✓	E01 (33)	Overlap (10%)
2	Toth's spell splits Xander into two personalities	E03:m11-12	(7) The demon hits Xander with light from a rod ...	✗	–	–
3	Monk tells Buffy that Dawn is the key	E05:m36-39	(34) He tells her that the key is a collection of energy put in human form, Dawn's form.	✓	E05 (34-35)	Overlap (31%)
4	A Queller demon attacks Joyce	E09:m32-33	(30) In Joyce's room, the demon falls from the ceiling ...	✓	E09 (28-30)	Overlap (12%)
5	Willow summons Olaf the troll	E11:m18-19	(17) Willow starts a spell, but Anya interrupts it ... (18) Accidentally, the spell calls forth a giant troll.	✗	–	–
6	Willow teleports Glory away	E13:m39-39	(34) ... before Willow and Tara perform a spell to teleport Glory somewhere else.	✓	E13 (34)	Overlap (63%)
7	Angel and Buffy in the graveyard	E17:m14-18	(13) At the graveyard, Angel does his best to comfort Buffy when she ...	✓	E17 (13-14)	Overlap (61%)
8	Glory sucks Tara's mind	E19:m24-27	(15) Protecting Dawn, Tara refuses, and Glory drains Tara's mind of sanity.	✓	E19 (14-15)	Overlap (74%)
9	Xander proposes Anya	E22:m16-19	(6) Xander proposes to Anya	✓	E22 (6)	2m44s

Impact of face id. We study the effect of face recognition (63% track recognition accuracy) by performing alignment with ground truth identities and obtain an average accuracy of 47.2%, roughly 6% better than what we achieve with automatic character identities (41.17%). After fusion with subtitles, the alignment based on ground truth ids (51.9%) is only 2.7% better than with automatic ids (49.2%).

Qualitative results. In Fig. 6, the alignment of three sample sentences from the plot synopsis of BF-02 is visualized. It is especially interesting to analyze the case for sentence BF-2:05 which contains the anchors Buffy and Giles. While we humans understand by reading the sentences that *Buffy* has to cancel her plans with *Riley* and thus annotate shots 73–82 (their discussion when they cancel), this is not easy to predict with the alignment technique. The alignment thus extends BF-2:04 which in fact contains *Riley* as a cue all the way to shot 78. Finally, since DTW3 cannot skip sentences, BF-2:05 is assigned to shots 79 and 80 after which BF-2:06 follows.

5.3 Retrieval Performance

We evaluate story-based retrieval on 62 queries and show a subset of the queries and their performance in Table 3. The retrieval finds relevant sentences for 53 (85%) of the 62 queries. For 24 (38%) queries we find the relevant sentence in the first position, for 43 (69%) in the top 5, and 48 (77%) in the top 10. The median position of the relevant result is 2. Note that rephrasing the queries can improve performance.

With respect to the position in the video, for 40 (64%) of 62 events the returned shots overlap with the ground truth. The remainder 13 events are located on average 3 minutes away from the actual depiction of the story across all 22 episodes.

6. CONCLUSION

We present a novel problem of searching for story events within large collections of TV episodes. To facilitate the retrieval, we propose to align crowd-sourced plot synopses with shots in the video. The alignment is formulated as an optimization problem and performed efficiently using dynamic programming. We evaluate the alignment against human annotations and show that 49% of the shots are assigned to the correct sentence. We also evaluate story-based retrieval on 15+ hours of video showing promising performance.

In the future, we intend to improve the alignment by using additional cues such as object detection and scene recognition. An open research question is an efficient alignment for non-sequential video descriptions (Fig. 4(mid)). We also would like to examine the alignment for other applications.

Acknowledgments

This work was realized as part of the Quaero Program, funded by OSEO, French State agency for innovation. The views expressed herein are the authors' responsibility and do not necessarily reflect those of OSEO.

7. REFERENCES

[1] Stanford CoreNLP. http://nlp.stanford.edu/software/.
[2] Whoosh - a Python full text indexing and search library. http://pypi.python.org/pypi/Whoosh.
[3] M. Bäuml, M. Tapaswi, and R. Stiefelhagen. Semi-supervised Learning with Constraints for Person Identification in Multimedia Data. In *CVPR*, 2013.
[4] T. Cour, C. Jordan, E. Miltsakaki, and B. Taskar. Movie/Script: Alignment and Parsing of Video and Text Transcription. In *ECCV*, 2008.
[5] M. Everingham, J. Sivic, and A. Zisserman. "Hello! My name is... Buffy" - Automatic Naming of Characters in TV Video. In *British Machine Vision Conference*, 2006.
[6] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every Picture Tells a Story : Generating Sentences from Images. In *ECCV*, 2010.
[7] J. L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382, 1971.
[8] B. Freiburg, J. Kamps, and C. Snoek. Crowdsourcing Visual Detectors for Video Search. In *ACM MM*, 2011.

BF-2:04. **Joyce** asks **Buffy** to take **Dawn** shopping for school supplies, but **Riley** reminds **her** they had already made plans.

BF-2:05. **Buffy** has to cancel so **she** can go work with **Giles**.

BF-2:06. **Giles** drives the **sisters** on their errands, having trouble with the automatic transmission in **his** new convertible.

Figure 6: Qualitative alignment results for 3 sentences from BF-02. The sentences from the plot synopsis are displayed at the top and *character names* are highlighted in bold and underline. This is followed by a list of representative images from each shot, with the shot number marked below. Arrow markers are used to indicate ground-truth assignment (green and above images) and the assignment predicted by DTW3 (red and below images). The ball indicates termination of the alignment while the arrow mark is used to show continuity.

[9] B. Fröba and A. Ernst. Face Detection with the Modified Census Transform. In *FG*, 2004.

[10] A. Gupta, P. Srinivasan, J. Shi, and L. S. Davis. Understanding Videos, Constructing Plots Learning a Visually Grounded Storyline Model from Annotated Videos Input. In *CVPR*, 2009.

[11] A. Habibian and C. Snoek. Video2Sentence and Vice Versa. In *ACM Multimedia Demo*, 2013.

[12] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning Realistic Human Actions from Movies. In *CVPR*, 2008.

[13] J. Law-To, G. Grefenstette, and J.-L. Gauvain. VoxaleadNews: Robust Automatic Segmentation of Video into Browsable Content. In *ACM Multimedia*, 2009.

[14] H. Lee, Y. Peirsman, A. Chang, N. Chambers, M. Surdeanu, and D. Jurafsky. Stanford's Multi-Pass Sieve Coreference Resolution System at the CoNLL-2011 Shared Task. In *Computational Natural Language Learning*, 2011.

[15] Y. Li, S.-H. Lee, C.-H. Yeh, and C.-C. Kuo. Techniques for Movie Content Analysis and Skimming. *IEEE Signal Processing Magazine*, 23(2):79–89, 2006.

[16] C. Liang, C. Xu, J. Cheng, W. Min, and H. Lu. Script-to-Movie : A Computational Framework for Story Movie Composition. *IEEE Transactions on Multimedia*, 15(2):401–414, 2013.

[17] C. S. Myers and L. R. Rabiner. A Comparative Study of Several Dynamic Time-Warping Algorithms for Connected Word Recognition. *Bell System Technical Journal*, 1981.

[18] Y. Peng and J. Xiao. Story-Based Retrieval by Learning and Measuring the Concept-Based and Content-Based Similarity. In *Advances in Multimedia Modeling*, 2010.

[19] J. Sang and C. Xu. Character-based Movie Summarization. In *ACM Multimedia*, 2010.

[20] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *MIR*, 2006.

[21] C. Snoek, B. Huurnink, L. Hollink, M. de Rijke, G. Schreiber, and M. Worring. Adding Semantics to Detectors for Video Retrieval. *IEEE Transactions on Multimedia*, 9(5):975–986, 2007.

[22] C. Snoek and M. Worring. Concept-Based Video Retrieval. *Foundations and Trends in Information Retrieval*, 4(2):215–322, 2009.

[23] C.-C. Tan, Y.-G. Jiang, and C.-W. Ngo. Towards Textually Describing Complex Video Contents with Audio-Visual Concept Classifiers. In *ACM Multimedia*, 2011.

[24] M. Tapaswi, M. Bäuml, and R. Stiefelhagen. "Knock! Knock! Who is it?" Probabilistic Person Identification in TV-Series. In *CVPR*, 2012.

[25] X. Wang, Y. Liu, D. Wang, and F. Wu. Cross-media Topic Mining on Wikipedia. In *ACM Multimedia*, 2013.

[26] C. Xu, Y.-F. Zhang, G. Zhu, Y. Rui, H. Lu, and Q. Huang. Using Webcast Text for Semantic Event Detection in Broadcast Sports Video. *IEEE Transactions on Multimedia*, 10(7):1342–1355, 2008.

[27] Y. Yusoff, W. Christmas, and J. Kittler. A Study on Automatic Shot Change Detection. *Multimedia Applications and Services*, 1998.

[28] H. Zaragoza, N. Craswell, M. Taylor, S. Saria, and S. Robertson. Microsoft Cambridge at TREC-13: Web and HARD tracks. In *Proc. TREC*, 2004.