Appeared in IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, June 2011

Extracting and Locating Temporal Motifs in Video Scenes Using a Hierarchical Non Parametric Bayesian Model

Rémi Emonet[1] Jagannadan Varadarajan[1,2] Jean-Marc Odobez[1,2]
[1]: Idiap Research Institute – CH-1920 Martigny, Switzerland
[2]: École Polytechnique Fédéral de Lausanne – CH-1015, Lausanne, Switzerland
{remonet,vjagann,odobez}@idiap.ch

Abstract

In this paper, we present an unsupervised method for mining activities in videos. From unlabeled video sequences of a scene, our method can automatically recover what are the recurrent temporal activity patterns (or motifs) and when they occur. Using non parametric Bayesian methods, we are able to automatically find both the underlying number of motifs and the number of motif occurrences in each document. The model's robustness is first validated on synthetic data. It is then applied on a large set of video data from state-of-the-art papers. We show that it can effectively recover temporal activities with high semantics for humans and strong temporal information. The model is also used for prediction where it is shown to be as efficient as other approaches. Although illustrated on video sequences, this model can be directly applied to various kinds of time series where multiple activities occur simultaneously.

1. Introduction

Mining recurrent temporal patterns in time series is an active research area. The objective is to find, with as little supervision as possible, the recurrent temporal patterns (or motifs) in time series. This general problem has its instance in computer vision where we would like to automatically extract activities from video sequences. A video sequence has the particularity of being "caused" by different activities acted by different persons or objects present in the scene.

Many other time series can present the same characteristic of being a fusion of multiple motifs. For example, we can consider the time series made of the overall electric and water consumption of a building. In such setting, we could observe motifs such as a short water consumption followed by short electric consumption (someone filling and then starting a boiler). We could also observe motifs like alternating water and electric consumptions for one hour (for a washing machine cycle). As multiple persons can live in the building (and one person can also do multiple tasks), multiple occurrences of these two motifs can occur at the same time and with no specific synchronization.

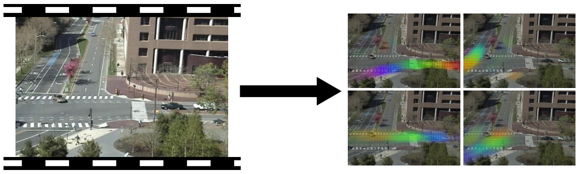

Figure 1. Task on video sequences. Without supervision, we want to extract recurring temporal activity patterns (4 are shown here). Time is represented using a gradient of color from violet to red. We call these patterns "motifs" in the article.

In the context of video sequences, the specific goal is to find activity patterns (e.g. car passing, pedestrian crossing) without supervision. This elementary task can be useful for applications like summarizing a scene, counting or detecting particular events or detecting unusual activities [8, 11, 7, 12]. More generally, this identification of temporal motifs and the instant at which they occur can be used as a dimensionality reduction method for (potentially supervised) higher level analysis.

In this paper, we present a model for finding temporal patterns (motifs) in time series. While selecting the number of motifs automatically, we also determine the number of times they occur in the data and when they occur. This model is well suited for any time series generated by non-synchronized concurrent activities as we illustrate by applying it to real video sequences.

2. Related Work

Recently, there has been an increased focus on discovering activity patterns from videos, especially in surveillance scenarios. These patterns are often called "activities" (or "motifs") in the existing literature. Although other paradigms can be successful as well (e.g. see an approach based on diffusion maps for instance [14]), topic models have shown tremendous potential in achieving this in an unsupervised fashion. Most of the existing topic model based methods propose to break the videos into clips of a few frames or seconds. Documents are created from these clips by quantizing pixel motion at different locations in the images. This approach was followed in [12, 11, 5], where activities are represented as static co-occurrences of words.

Activities in a video are by nature, temporally ordered. Therefore, following the exchangeability assumption and representing each action as a bag-of-words [12, 11] results in loosing the temporal dependencies among the words. Several attempts have been made to incorporate temporal information in topic models, starting from the work done in text processing [2, 13]. Following these lines, methods in [4] improve by modeling the sequence of scene behaviors as a markov model, but with a pre-determined fixed set of topics. While the temporal order is imposed at the global scene level, the higher level of the hierarchy, the activity patterns are still modeled as static distributions over words. The methods proposed in [7, 3, 10] complement visual words with their time stamps to recover temporal patterns. While this method can be useful when clips are aligned to recurring cycles like traffic signals (as this was done manually in [3]), it gives poor results in general cases where such alignment is not done a priori [7]. A more general approach was proposed in [10], wherein motifs and their starting times are jointly learnt, requiring no manual alignment of clips. However, the model is not fully generative and requires setting various parameters like the number of topics.

One of the main challenges in topic model based activity modelling is model selection, that is, the automatic estimation of the number of topics. Non-parametric Bayesian methods such as Hierarchical Dirichlet Process [9] allows to have in theory an infinite number of topics and in practice to select this number. Such a model was explored for discovering static topics in [12]. Similarly, [5] uses the HDP and infinite state HDP-HMM paradigms of [9] to identify temporal topics and scene level rules. Unfortunately, in practice, only a single HMM was found for each of the four tested scenes, meaning that temporal ordering was still dominantly modeled as the global scene level using a set of static activity distributions, similarly to what was done in [4].

Our paper differs significantly from previous work. Our aim is to find both motifs with strong explicit temporal information and when they appear in the temporal documents. The main contributions of this work are: we adopt a Non-parametric Bayesian approach to automatically determine the number of topics shared by the documents and also when they appear in each temporal document. To this end, we derive the Gibbs sampler for the joint inference of the topics and their start times. We propose a method to predict the future occurrences by inferring the start times of topics on partial occurrences. We finally compare our prediction method to a more traditional HMM based prediction and validate the effectiveness of our method on a large set of video data provided by [12, 4, 10, 5].

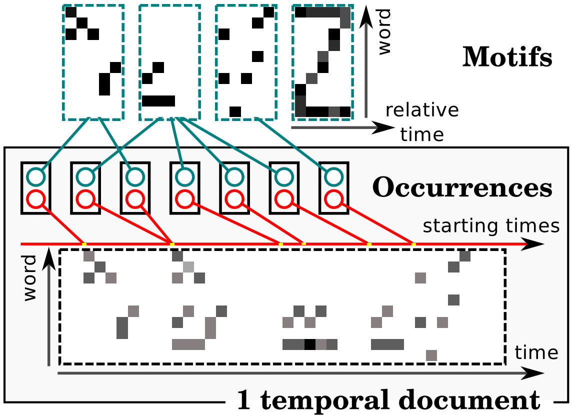

Figure 2. Schematic generative model. A temporal document is made of words counts at each time instant. Each document is composed of a set of occurrences each having one motif and a location in time (starting time). The motifs are shared by the occurrences within and across documents.

3. Approach Overview

The input data of our method is a set of temporal documents (possibly a long single one) as defined in section 1. This observed document is defined as a table of counts (or table of amount of presence): for each word in a defined vocabulary and at each time instant, the table reflects the amount of presence of this word at this time instant. Our approach is depicted in figure 2 where each document is represented as a set of "motif occurrences" (e.g. 7 of them in Fig. 2). Each occurrence is defined by a starting time instant and a motif. Motifs are shared by different occurrences within and across documents.

In our model, we extensively use Dirichlet Processes (DP). A DP is a non-parametric Bayesian process to model infinite mixture model. The term "non-parametric" refers to the fact that the model grows in function of the observed data. Dirichlet processes are often used to determine automatically the number of relevant elements in a mixture model (e.g. number of topics or number of gaussians). A DP is an infinite mixture but observations from a DP most probably tend to cluster on some limited elements of the mixture.

We use two levels of DP in our approach. At a lower level, within each document, we model the occurrences using a DP: observation then cluster around an automatically determined number of occurrences. At an higher level, we model the motifs using a DP: occurrences within and across documents then cluster around an automatically determined number of motifs.

4. Model

As introduced in section 3, our model relies on Dirichlet Processes (DP) to discover the temporal motifs, their number, and find their occurrences. We will thus start by introducing DP and then describe our model with more details.

4.1. Hierarchical Dirichlet Processes (HDP)

Dirichlet processes are a non-parametric approach to model arbitrary distributions over data points using mixture models having a potentially infinite number of mixtures. Hierarchical Dirichlet Processes [9] have been introduced as a way to generalize DP by modeling distributions in different groups of data (the documents) using DP and allowing these document specific mixture models to share the same mixture components.

In the context of this section, we will assume that the data are words belonging to a vocabulary, and that the mixture components are topics represented by multinomial distributions other these words.

Below we summarize the main points of the HDP. The reader is referred to [9] for a comprehensive description.

Hierarchical Dirichlet process. The HDP generative process is presented in Fig. 3. It can be described as follows [9]:

$$\begin{array}{ll} G^0 \sim DP(\gamma, H) & G_j \sim DP(\alpha, G^0) \\ \theta_{ji} \sim G_j & x_{ji} \sim \text{Mult}(\theta_{ji}) \end{array} \quad (1)$$

As can be seen, it uses Dirichlet Processes at two levels. In the first level, the DP generates a distribution G^0 defined as an infinite mixture of topic atoms ϕ_l drawn from the base distribution H, where the mixture weights are generated using the "GEM" stick-breaking construction [9]. Thus, the Dirichlet Process can also be written as:

$$G^0 = \sum_{l=1}^{\infty} \beta_l \delta_{\phi_l} \text{ with } \beta \sim GEM(\gamma) \text{ and } \phi_l \sim H \quad (2)$$

Note that it follows from this definition that a draw from G^0 produces one of the atom ϕ_l. G^0 contains the list of all topics ϕ_l that are shared by all documents. The concentration γ is a prior parameter on the topic occurrence distribution that will influence the actual (finite) number of topics that will be recovered during the inference process[1].

The second Dirichlet Process $G_j \sim DP(\alpha, G^0)$ in the HDP allows to define the mixture of topics in each document j. Since the base distribution G^0 is itself a DP, the distribution G_j will actually be a mixture defined over the same set of atoms as in G^0. This is this property of DP that allows documents to share the same topics. In practice however, only a subset of the topics in G^0 are actually active in G_j and used to generate the words occurring in the document. The document specific topic weights π_j are given by:

$$G_j = \sum_{l=1}^{\infty} \pi_{jl} \delta_{\phi_l} \text{ with } \pi_j \sim DP(\alpha, \beta) \quad (3)$$

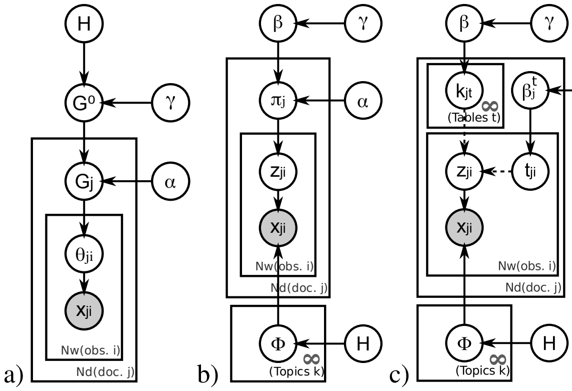

Figure 3. Three representations of HDP [9]. We can develop Dirichlet processes (DP) as a stick breaking process and some draws from the base distribution. a) most compact representation; b) developing the higher level DP, c) developing both DP.

where β is the base distribution of topic weights in G^0 (cf Eq.2). This representation is illustrated in Fig. 3b. The DP process in Eq.3 can be further decomposed using a GEM process and a set of mixtures elements. It is illustrated in Fig. 3c and corresponds to the generative process:

$$\begin{array}{ll} k_{jt} \sim \beta & \beta_j^t \sim GEM(\alpha) \\ t_{ji} \sim \beta_j^t & z_{ij} = k_{jt_{ji}} \text{ and } x_{ji} \sim \text{Mult}(\phi_{z_{ji}}) \end{array} \quad (4)$$

which reflects the analogy with the Chinese Restaurant Franchise Process[2] [9]. There, customers/words x_{ji} in a restaurant/document j sit at different tables, and at each table t in this restaurant the meal/topic k_{jt} is served. In HDP, the global GEM distribution β is used to draw the meals k_{jt} associated with each table t, while the document specific GEM β_j^t is used to draw the tables t_{ji} at which each customers x_{ji} sits. Hence, given the table t_{ji} at which he sits, a customer's meal z_{ji} is automatically determined given the set of k_{jt}. Notice that, although it is not necessary, we used dashed arrows in the graphical model to represent such deterministic relations (the conditional probability is a Dirac distribution).

4.2. Proposed Model

Our goal is to automatically infer a set of motifs (temporal activity topics) from a set of documents containing time-indexed words.

More precisely, let us define a document j as a set of observations $(w_{ji}, at_{ji})_{i=1 \ldots N_j}$, where w_{ji} is a word belonging to a vocabulary $\mathcal{V}$ describing a localized spatio-temporal activity in the image (how we get these words is defined in the Experiment Section), and at_{ji} is the absolute time instant at which the word occurs.

[1] Although the number of topics can potentially be infinite, in practice only a few of them have a significant enough weight.

[2] Note that the β and β_j^t distributions resulting from the GEM process can be seen as distributions over the infinite set of positive integers.

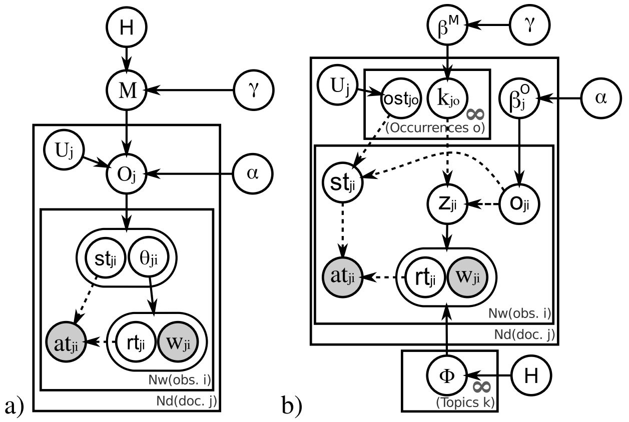

Figure 4. Proposed model a) with DP compact notation; b) with developed Dirichlet processes (using stick-breaking convention at both levels). Dashed arrows represents deterministic relations (conditional distributions are a Dirac).

Similarly, let us define our motifs as spatio-temporal probabilistic maps. More precisely, if ϕ_l denotes a motif, then $\phi_l(rt, w)$ denotes the probability that the word w occurs rt relative time steps after the start of the motif.

Our goal is to infer the set of motifs from the documents. As discussed previously, this require to simultaneously infer the starting times of each motif occurrence in the document. As it is difficult to fix the number of motifs before hand, we use a DP to allow the learning of a variable number of models from the data. Similarly, we use a second DP to model all motif occurrences as we don't know their number in advance.

Our generative model is thus defined using the graphical models presented in Fig. 4. Fig.4a depicts our model using the compact Dirichlet processes notation as done for HDP in Fig. 3a, whereas Fig.4b depicts the developed notation (cf Fig. 3c). Notice that in these drawings, two variables in an ellipse form a couple, indicating that they are tied by their plate index.

The equations associated with Fig. 4a are as follows:

$$\begin{aligned}
M &\sim DP(\gamma, H) \text{ with } H = Dir(\eta) && (5)\\
O_j &\sim DP(\alpha, (U_j, M)) && (6)\\
(st_{ji}, \theta_{ji}) &\sim O_j && (7)\\
(rt_{ji}, w_{ji}) &\sim \text{Mult}(\theta_{ji}) && (8)\\
at_{ji} &= st_{ji} + rt_{ji} && (9)
\end{aligned}$$

where deterministic relations are denoted with "=". As with the standard HDP, the first DP level generates our list of motifs in the form of an infinite mixture M. Each of the motif is drawn from H, defined as a Dirichlet of parameter η (a table of the size of a motif; see below how we set it).

However, contrary, to the standard HDP, this set of atoms is not only shared across documents, but also across motif occurrences using the DP at the second level. More precisely, the document specific distribution O_j is not defined as a mixture over motifs, but as an infinite mixture over (start time × motifs) occurrence atoms (cf Fig. 2), since the base distribution is defined by (U_j, M). Each of the atoms is thus a couple (ost_l, ϕ_l), where $ost_l \sim U_j$ is the occurrence starting time drawn from U_j, a uniform distribution over the set of possible motif starting times in the document j, and $\phi_l \sim M$ is one of the topic drawn from the mixture of motifs.

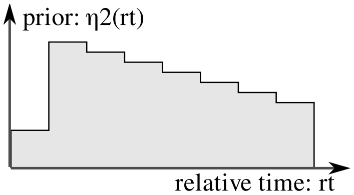

Figure 5. Motif prior over the relative time occurrence of words rt (for a 8 time steps motif). The steep increasing ramp at the first two time steps ensures that real activities are captured from their start, (i.e. words occurring prior to the start are not consistent with the activity), while the decreasing ramp ensures that most of the activity concentrates towards the beginning of the motif.

Words (w_{ji}) are then generated by repeatedly sampling a motif occurrence (Eq. 7), using the obtained motif θ_{ji} to sample the word w_{ji} and its relative time in the motif rt_{ji} (Eq. 8), from which, using the sampled starting time st_{ji}, the word absolute time occurrence at_{ji} can be deduced (Eq. 9).

The fully developed model in Fig. 4b helps to better understand the generation process and the inference. The corresponding equations can be written as:

$$\begin{aligned}
\beta^M &\sim GEM(\gamma) \qquad \phi_l \sim H && (10)\\
\beta^o_j &\sim GEM(\alpha) && (11)\\
ost_{jo} &\sim U_j \text{ and } k_{jo} \sim \beta^M && (12)\\
o_{ji} &\sim \beta^o_j && (13)\\
z_{ji} &= k_{jo_{ji}} \text{ and } st_{ji} = ost_{jo_{ji}} && (14)\\
(rt_{ji}, w_{ji}) &\sim \text{Mult}(\phi_{z_{ji}}) && (15)\\
at_{ji} &= st_{ji} + rt_{ji} && (16)
\end{aligned}$$

The main difference with the compact model is that the way motif occurrences are generated is explicitly represented. Occurrences are the analog of the tables in HDP: the global GEM distribution over motifs β^M and U_j are used to associate motif indices k_{jo} and starting times ost_{jo} to each occurrence (Eq. 12), while the document specific GEM β^o_j is used to sample the occurrence associated to each word (Eq. 13), from which generating the observations can be done as presented above (Eq. 14 to 16).

Setting time dependent prior on motifs. The parameter η

(a table of the size of the motifs) defines the Dirichlet prior $H = Dir(\eta)$ from which the motifs ϕ_k (defined as multinomials over (w, rt)) are drawn. The normalized vector $\eta' = \frac{\eta}{\|\eta\|}$ represents the expected values for the multinomial coefficients, whereas the strength $\|\eta\| = \sum_{w,rt} \eta(w, rt)$ influences the variability around this expectation. A larger norm $\|\eta\|$ results in lower variability. Often, a uniform prior over words is used, and only the prior strength is changed.

Since η' can be assimilated to parameters of a multinomial, we will set it as:

$$\eta'(w, rt) = \eta_1(w|rt)\eta_2(rt). \tag{17}$$

in which we define the word probabilities η_1 for a given rt to be uniform, and the prior η_2 on rt according to Fig. 5. The impact of this prior during inference will be three-fold: first, the steep increasing ramp in the first two time steps ensures that real activities are captured from their start, i.e. there is no word before or at the start that occurs consistently with the activity. Second, it will avoid the discovery of motifs with no activity at all at the beginning. Third, when seeking for longer motifs, it will reduce the learning of spurious co-occurrences by allowing a graceful dampening of word occurrences at the end of the motifs unless their co-occurrence with words appearing in the first part of the motif is strong enough.

5. Inference

Compared to plain HDP, our inference is more complicated given the added starting time to our occurrences (HDP tables). Given the model presented in section 4, we use a Gibbs sampling and sample over:

- o_{ji}: association of observations to occurrences
- k_{jo}: association of occurrences to motifs
- ost_{jo}: starting times of occurrences

Using a Dirichlet prior as H (which is conjugate to the observations), we can integrate out the motifs themselves as in standard HDP. Due to space constraints, more detailed equations used in the Gibbs sampling process are provided as additional material [1]. In this section we summarize the main elements in the Gibbs sampler.

We recall that a Dirichlet Process (DP) can also be defined using the a Chinese restaurant process. For example, we can consider a DP of concentration γ and base distribution H. In the Chinese restaurant definition, given a set of previous draws from this DP, a new draw is obtained by considering two possible cases. Firstly, the new draw can be exactly one of the previous draws, this happens with a probability proportional to the number of previous draws having this exact value. Secondly, the new draw can be drawn directly from H, this happens with a probability proportional to the concentration γ. This Chinese restaurant process is highly used in the derivation of Gibbs sampling equations.

Sampling o_{ji} (for a given observation i in document j) requires to consider two cases: either the observation will be re-affected to an existing occurrence or a new occurrence will be created for it.

The probability of affecting an observation to a particular existing occurrence is proportional to two quantities. The first quantity, because of the Chinese restaurant process on the occurrences, is the number of observations that are associated with the considered occurrence. The second quantity comes from the likelihood of the considered observation given its virtual affectation to the considered occurrence. From the occurrence starting time and the observation time, we can calculate the relative time rt_{ji} of the observation in the motif. Considering the prior H and all observations (across documents) affected to the occurrence motif, we can compute the likelihood of the considered observation with its relative time.

The other option is to create a new occurrence for the observation. Because of the Chinese restaurant process, it will be proportional to α. This probability of creating a new occurrence is also proportional to the likelihood of the considered observation under the hypothesis that it is associated to a new random occurrence. To evaluate this last probability, we need to consider the expected value over all possible starting times and all possible motifs for the new occurrence. With a uniform prior on the starting times, we manage to integrate over the starting times. Considering all possible motifs is more difficult: here again we have a DP and, the motif can be either an existing one (with a probability proportional to the number of occurrences across documents for this motifs) or a new motif drawn from H with a probability γ. Given our Dirichlet prior H, we manage to integrate over the new motifs drawn from H.

About the α concentration parameter. In HDP, the α controls reaffectation of an observation to a new table versus any existing table. In our model, α controls reaffectation of an observation to a new occurrence versus only the occurrences that can explain this observation: if an occurrence is too far from an observation then the observation cannot be reaffected to this occurrence.

Phrased differently, in HDP α controls the number of topics used in a document. In our model α does not control the number of occurrences in a document but rather the average number of overlapping occurrence at a particular time instant in the document. The consequence is that α can be set independently of the document length and that it takes relatively small values.

Sampling k_{jo} and ost_{jo} . We resample independently each k_{jo} and, to encourage faster alignment of motifs, we resample ost_{jo} by groups: we consider in turn each group made of all occurrence having a particular motif. Due to space constraints, we redirect the interested reader to additional material [1] for details on this sampling.

6. Experiments

Here, we present results on synthetic and video data. More illustrations are available in additional material [1].

6.1. Validation on synthetic documents

To validate our model, we apply it on synthetic temporal documents. We first randomly generate a set of 5 motifs with 25 words and 5 time steps. Then, we generate documents of 100 time steps, each containing 40 random occurrences and 20 random observations. From this we obtain dense documents where occurrences are overlapping almost everywhere.

Number of motifs. Our model properly finds the number of motifs and fully recovers the original motifs (see [1]). Due to random co-occurrences in these dense documents, sometimes an additional motif containing multiple other motifs is also recovered.

Motif duration. In our model, we need to provide a maximum motif duration for recovered motifs. Considering the real motifs are of 5 time steps, if we ask for even shorter motifs, then we recover multiple motifs to explain a single real motif. If we ask for longer motifs, we observe that our prior on the relative time (see section 4) is effectively encouraging the real motifs to start at the beginning of the recovered motif. With a uniform prior we get the real motif at a random position in the recovered motif. Such randomness makes the interpretation more difficult as the recovered motifs start with an unknown duration of almost no activity.

Noise. We also tested the effect of noise on our documents: as in [10], we add some observations drawn uniformly in the document (both a word and an absolute time are drawn). In [10] it is reported that they recover the original motifs with added noise. Interestingly, in our case, as the model automatically selects the number of motifs, it creates one or more noisy motifs to explain the noise present in the data. As the model explicitly creates noise motifs, the recovered non-noise motifs appear with much less noise. This clearly illustrates the model's freedom in selecting the most appropriate number of motifs under different situations.

6.2. Experiments on video data

We consider different videos taken from static cameras and use our model to discover recurrent motion patterns in the scenes. In the body of the article (Fig. 6 and 7) we show only 3 different scenes: the MIT video from [12], the UQM roundabout video from [6] and the far-field video from [10] (Fig. 6). We also experimented on other videos (e.g. from [4] and [5]): results are shown in [1].

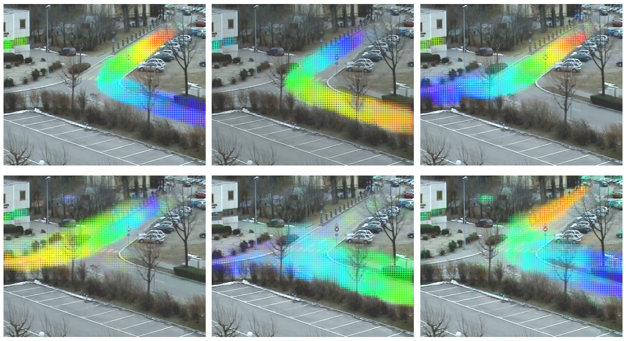

Figure 6. (best viewed in color) Top 6 recovered motifs, explaining more than 95% of the data. Time is represented using a gradient of color from violet to red. Displayed motifs are all composed of 11 time instants (seconds).

From Videos to Temporal Documents. To create a temporal document from a video, we first extract low-level visual features. We temporally downsample all the videos to 5 frames per second and compute the optical flow features at sampled locations of a fine spatial grid using a modified version of the opencv KLT code. The optical flow vectors are then quantized into 8 cardinal directions. A post-processing step is then applied to only keep the words that appear sufficiently often and doing so prune the areas where nothing happens in the images. After this process we get around 20k different words which define our low-level codebook.

Rather than using these low-level words directly in our model, as done in most works, we first apply a dimensionality reduction using Probabilistic Latent Semantic Analysis (PLSA). More precisely, documents are created by counting the number of occurrences of the low-level words in each second of the video. The PLSA algorithm is then applied on these documents to learn a set of localized activity topics. We ask for a conservative number of topics (between 50 and 100 depending on the video complexity) in order to keep a representation with a fine enough spatial resolution.

Finally, the estimated PLSA topics are then used to generate the temporal documents that we feed into our model by considering each PLSA topic z_{plsa} as a word w of our motif model. This is achieved by applying the PLSA inference (assuming known topics) at each time step at (we used a temporal resolution of one second) on low-level word document d_{at} to obtain the posterior estimate of $p(z_{plsa}|d_{at})$. The amount of words $n(w, at)$ used as input in our algorithm is then defined as: $n(w, at) \propto N_{at} \cdot p(z_{plsa}|d_{at})$, where N_{at} denotes the number of low-level words in the document d_{at}.

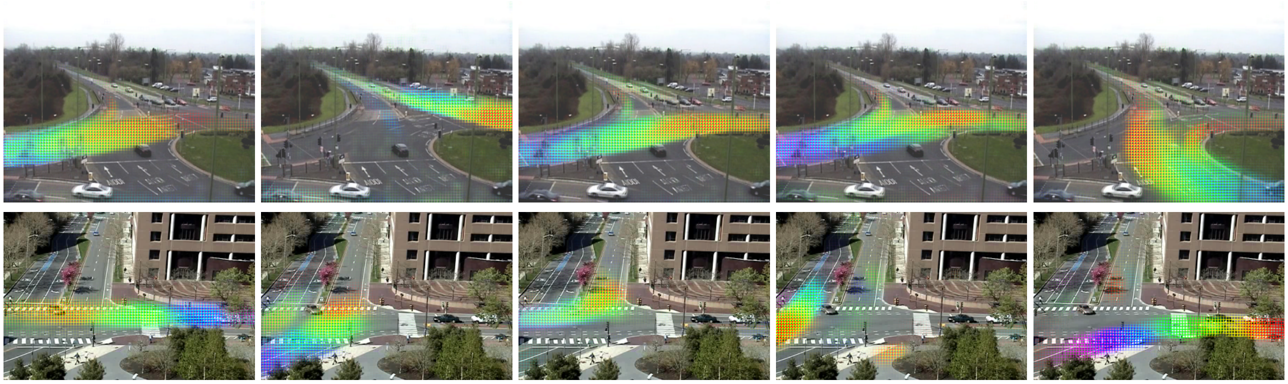

Figure 7. (best viewed in color) Example of the top 5 recovered motifs from different scenes. Time is represented using a gradient of color from violet to red. Displayed motifs are all composed of 6 time instants (second). See additional material [1] for more examples.

Recovered Motifs. We run the inference of our model on different video datasets to retrieve recurrent activities as motifs. A recovered motif is a table providing the probability that a word occur at a relative time instant with respect to the beginning of the motif. Since, as introduced above, each word corresponds to the response of a PLSA topic z_{plsa}, we can backproject the set of locations where it is active in the image plane[3]. Subsequently, to visualize the content of each motif, the word probabilities for each relative time instants rt are backprojected into the image plane to obtain activity images I_{rt} from which a short video clip can be generated (e.g. animated gifs).

We use a static color coded representation to show examples in this paper. Each time instant is assigned a color (from blue to red) and superimposed in a single image. This representation is more compact than showing all images but suffers from "occlusions" when motion is slow (e.g. blue is occluded by cyan and green).

Fig. 6 and 7 show the most probable recovered motifs for three different datasets. Other datasets can be found in additional material [1]. The results are interesting and the motifs recovered by our method actually correspond to real activities. Analyzing the results, we see that they underline the following key behaviors of our model.

Our model properly captures multi-object patterns. The motifs capture, for instance, co-occurring car motions as in Fig. 7 (e.g. second row, second column) and trams blocking cars in ETHZ (see [1]). Also, the training sets are small (under one hour) and some motifs capture fortuitous co-occurrences or specific events (e.g. the last motif in Fig. 6 corresponds to a few huge trucks).

Motif duration. Our model behaves well when changing the maximum motif duration parameter. If this parameter is shorter than the real durations of the activities, each activity is cut into multiple motifs [1]. When the motifs are long enough, each activity is captured by a motif as in Fig. 6 and 7. Also, thanks to the temporal prior (Fig 5), the activities are properly aligned at the begin of the motifs. We even recover the full traffic lights cycle (if any) when asking for motifs of 2 minutes as with the UQM dataset (see [1]).

Execution speed. Our model is sensible to variations of execution speed of activities: only small variations are usually captured by a motif. Often, multiple motifs capture the "same activity" but at different speed. In Fig. 7 for example, this explains why the motifs at column c1 and c3 of the first row are almost the same but differ by their duration (color range): c1 captures cars starting and crossing the scene slowly while column c3 captures faster cars. Note that our model could account for this problem by adding a speed variable in each occurrence.

Using Model for Prediction. Our method captures meaningful motifs but we also want to validate it quantitatively. We design a prediction task in order to be able to compare our approach to other ones.

We consider two different datasets: the MIT dataset (second row in Fig. 7) and Far field dataset (Fig. 6). The former is a 4 road junction scene controlled by periodic traffic lights. The latter is an uncontrolled setting with activities of large temporal variations. In both cases, 80% of the data (around one hour) are used for training the considered models and 20% are used for evaluation.

For the evaluation, we take a sliding window of 30 time instants and use the first 29 time instants to predict the last one. The prediction is normalized (if needed) and compared to the real observations using a Bhattacharyya similarity. The normalization step is necessary to be able to compare methods (e.g. Hidden Markov Models (HMM)) but works to the disadvantage of our method that could predict not only a distribution but also an amount of activity. We aver-

[3] Note that the PLSA topic contain more than the image location, i.e. their low-level word distribution $p(w_{lowlevel}|z_{plsa})$ provide information about the motion direction distribution. However, for visualization purposes, only the locations are shown.

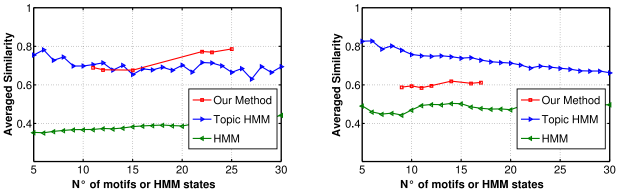

Figure 8. Accuracy on a prediction task (left: far field video, right: MIT video). In spite of the task being unfriendly to our method (see body of the article), we get good results.

age the similarity measure over all possible test windows.

We compare our method to two other ones. The first is based on an HMM learned in an unsupervised way by maximizing the training data likelihood. The second is a more sophisticated approach in line with [4], wherein the Markov chain runs between the global behavior states. For this, we first apply PLSA (with n topics) by considering each time instant as a document to obtain a topic distribution $p(z|d)$ for each time instant. An HMM (with n states) is then learnt using these topic distributions as observations. The HMM states learnt in this method are distinct scene level activity interactions or behaviors and the Markov chain gives the temporal dependencies among the states. We refer to this method as *Topic HMM*.

Fig. 8 shows the predictive accuracy of our method and the HMM-based methods. Note that the x-axis in the plots represents either the number of HMM states or the number of motifs selected by our method during its several runs by varying the topic duration. While the HMM state count is manually varied, our method automatically selects the number of motifs and hence the variation in motifs count is limited.

The predictive performance of our method is comparable to the Topic HMM-based method. In the MIT data, our method marginally lags behind the Topic HMM method. This is mainly due to traffic lights: the scene goes through distinct global behavior states [1] which are explicitly modeled in Topic HMM whereas our method does not have any prior on the sequences of occurrences (Fig. 2). In the more uncontrolled case of the Far-Field scene, we find that our method performs better than the Topic HMM. This clearly demonstrates that our method is capable of extracting motifs with high semantic content without compromising its predictive capabilities.

7. Conclusions

This paper introduced a new model capable of automatically finding recurrent temporal patterns (motifs) in time series. The model automatically and jointly finds: 1) the shape and number of motifs common to a set of temporal documents, and 2) how many times and when these motifs appear in each document. The model has been validated on synthetic data and applied to find recurrent activities in videos. On video data, the model extracts motifs that capture meaningful activities with a strong temporal semantic.

This model is not limited to video data and can be applied on various time series. The design of this model makes it most suitable for cases where the observed time series are the superposition of multiple unsynchronized activities.

Acknowledgements

The authors gratefully acknowledge the financial support from the Swiss National Science Foundation (Project: FNS-198,HAI) and from the 7th framework program of the European Union project VANAHEIM (248907) under which this work was carried out.

References

[1] Paper webpage. http://www.idiap.ch/paper/2053.

[2] D. Blei and J. Lafferty. Dynamic topic models. In *Proc. of the 23rd Int. Conference on Machine Learning*, 2006.

[3] T. A. Faruquie, P. K. Kalra, and S. Banerjee. Time based activity inference using latent dirichlet allocation. In *British Machine Vision Conference*, London, UK, 2009.

[4] T. Hospedales, S. Gong, and T. Xiang. A markov clustering topic model for mining behavior in video. In *International Conference in Computer Vision*, Kyoto, Japan, 2009.

[5] D. Kuettel, M. D. Breitenstein, L. V. Gool, and V. Ferrari. What's going on? discovering spatio-temporal dependencies in dynamic scenes. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, 2010.

[6] J. Li, S. Gong, and T. Xiang. Scene segmentation for behaviour correlation. In *ECCV*, France, 2008.

[7] J. Li, S. Gong, and T. Xiang. Discovering multi-camera behaviour correlations for on-the-fly global activity prediction and anomaly detection. In *IEEE International Workshop on Visual Surveillance*, Kyoto, Japan, 2009.

[8] B. Luvison, T. Chateau, P. Sayd, Q.-C. Pham, and J.-T. Lapresté. An unsupervised learning based approach for unexpected event detection. In *VISAPP*, 2009.

[9] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

[10] J. Varadarajan, R. Emonet, and J. Odobez. Probabilistic latent sequential motifs: Discovering temporal activity patterns in video scenes. In *Procedings of the British Machine Vision Conference*, Aberystwyth, 2010.

[11] J. Varadarajan and J. Odobez. Topic models for scene analysis and abnormality detection. In *ICCV-12th International Workshop on Visual Surveillance*, Kyoto, Japan, 2009.

[12] X. Wang, X. Ma, and E. L. Grimson. Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *IEEE Trans. PAMI*, 2008.

[13] X. Wang and A. McCallum. Topics over time: A non-markov continuous-time model of topical trends. In *In Conference on Knowledge Discovery and Data Mining (KDD)*, 2006.

[14] Y. Yang, J. Liu, and M. Shah. Video scene understanding using multi-scale analysis. In *ICCV*, Kyoto, Japan, 2009.