Random Field Topic Model for Semantic Region Analysis in Crowded Scenes from Tracklets

Bolei Zhou[1], Xiaogang Wang[2,3], and Xiaoou Tang[1,3]
[1]Department of Information Engineering, The Chinese University of Hong Kong
[2]Department of Electronic Engineering, The Chinese University of Hong Kong
[3]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
zhoubolei@gmail.com, xgwang@ee.cuhk.edu.hk, xtang@ie.cuhk.edu.hk

Abstract

*In this paper, a Random Field Topic (**RFT**) model is proposed for semantic region analysis from motions of objects in crowded scenes. Different from existing approaches of learning semantic regions either from optical flows or from complete trajectories, our model assumes that fragments of trajectories (called tracklets) are observed in crowded scenes. It advances the existing Latent Dirichlet Allocation topic model, by integrating the Markov random fields (MRF) as prior to enforce the spatial and temporal coherence between tracklets during the learning process. Two kinds of MRF, pairwise MRF and the forest of randomly spanning trees, are defined. Another contribution of this model is to include sources and sinks as high-level semantic prior, which effectively improves the learning of semantic regions and the clustering of tracklets. Experiments on a large scale data set, which includes $40,000+$ tracklets collected from the crowded New York Grand Central station, show that our model outperforms state-of-the-art methods both on qualitative results of learning semantic regions and on quantitative results of clustering tracklets.*

1. Introduction

In far-field video surveillance, it is of great interest to automatically segment the scene into semantic regions and learn their models. These semantic regions correspond to different paths commonly taken by objects, and activities observed in the same semantic region have similar semantic interpretation. Some examples are shown in Figure 1 (A). Semantic regions can be used for activity analysis in a single camera view [21, 9, 10, 24, 20] or in multiple camera views [13, 11, 23] at later stages. For example, in [21, 9, 10, 24] local motions were classified into atomic activities if they were observed in certain semantic regions and the global behaviors of video clips were modeled as distributions of over atomic activities. In [20], trajectories of objects were classified into different activity categories according to the semantic regions they passed through. In [13, 11], activities in multiple camera views were jointly modeled by exploring the correlations of semantic regions in different camera views. Semantic regions were also used to improve object detection, classification and tracking [5].

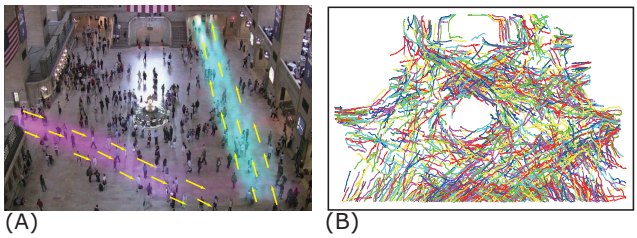
(A) (B)

Figure 1. (A) The New York Grand Central station. Two semantic regions learned by our algorithm are plotted on the background image. They correspond to paths of pedestrians. Colors indicate different moving directions of pedestrians. Activities observed on the same semantic region have similar semantic interpretation such as "pedestrians enter the hall from entrance *a* and leave from exit *b*".(B) Examples of tracklets collected in the scene. The goal of this work is to learn semantic regions from tracklets.

Generally speaking, the approaches of learning semantic regions are in two categories: local motion based (such as optical flows) [21, 11, 9, 10, 4] and complete trajectories of objects [14, 6, 22, 20] based. Both have some limitations. Without tracking objects, discriminative power of local motions is limited. The semantic regions learned from local motions are less accurate, tend to be in short range and may fail in certain scenarios. The other type of approaches assumed that complete trajectories of objects were available and semantic regions were estimated from the spatial extents of trajectory clusters. However this assumption is hard to be guaranteed due to scene clutters and tracking errors, thus the learned semantic regions are either oversegmented or improperly merged.

1.1. Our approach

We propose a new approach of learning semantic regions from tracklets, which are a mid-level representation between the two extremes discussed above [1]. A tracklet is a fragment of a trajectory and is obtained by a tracker within a short period. Tracklets terminate when ambiguities caused by occlusions and scene clutters arise. They are more conservative and less likely to drift than long trajectories. In our approach, a KLT keypoint tracker [18] is used and tracklets can be extracted even from very crowded scenes.

A Random Field Topic (**RFT**) model is proposed to learn semantic regions from tracklets and to cluster tracklets. It advances the Latent Dirichlet Allocation topic model (LDA) [2], by integrating MRF as prior to enforce the spatial and temporal coherence between tracklets during the learning process. Different from existing trajectory clustering approaches which assumed that trajectories were independent given their cluster labels, our model defines two kinds of MRF, pairwise MRF and the forest of randomly spanning trees, over tracklets to model their spatial and temporal connections.

Our model also includes sources and sinks as high-level semantic prior. Although sources and sinks were explored in existing works [14, 17, 22] as important scene structures, to the best of our knowledge they were not well explored to improve the segmentation of semantic regions or the clustering of trajectories. Our work shows that incorporating them in our Bayesian model effectively improves both the learning of semantic regions and the clustering of tracklets.

Experiments on a large scale data set include more than $40,000$ tracklets collected from the New York Grand Central station, which is a well known crowded and busy scene, show that our model outperforms state-of-the-art methods both on qualitative results of learning semantic regions and on quantitative results of clustering tracklets.

1.2. Related works

Wang et al.[21] used hierarchical Bayesian models to learn semantic regions from the temporal co-occurrence of optical flow features. It worked well for traffic scenes where at different time different subsets of activities were observed. However, our experiments show that it fails in a scene like Figure 1 (A), where all types of activities happen together most of the time with significant temporal overlaps. In this type of scenes, the temporal co-occurrence information is not discriminative enough. Some approaches [11, 9, 10, 4] segmented semantic regions by grouping neighboring cells with similar location or motion patterns. Their segmentation results were not accurate and tended to be in short ranges.

[1]Optical flows only track points between two frames. The other extreme is to track objects throughout their existence in the scene.

Many trajectory clustering approaches first defined the pairwise distances [8, 1] between trajectories, and then the computed distance matrices were input to standard clustering algorithms [6]. Some other approaches [25, 16] of extracting features from trajectories for clustering were proposed in recent years. Semantic regions were estimated from the spatial extents of trajectory clusters. It was difficult for those non-Bayesian approaches to include high-level semantic priors such as sources and sinks to improve clustering. Wang et al. [20] proposed a Bayesian approach of simultaneously learning semantic regions and clustering trajectories using a topic model. This work was relevant to ours. However, in their generative model, trajectories were assumed to be independent given their cluster assignments and the spatial and temporal connections between trajectories were not modeled. It worked well in sparse scenes where a large portion of trajectories were complete, but not for crowded scenes where only tracklets can be extracted reliably. It did not include sources and sinks as prior either.

Tracklets were explored in previous works [3, 12] mainly for the purpose of connecting them into complete trajectories but not for learning semantic regions or clustering trajectories. Our approach does not require first obtaining complete trajectories from tracklets.

In recent years, topic models borrowed from language processing were extended to capture spatial and temporal dependency to solve computer vision problems. Hospedales et al. [4] combined topic models with HMM to analyze the temporal behaviors of video clips in surveillance. A temporal order sensitive topic model was proposed by Li et al. [11] to model activities in multiple camera views from local motion features. Verbeek et al. [19] combined topic models with MRF for object segmentation. Their model was relevant to ours. In [19], MRF was used to model spatial dependency among words within the same documents, while our model captures the spatial and temporal dependency of words across different documents. Moreover, our model has extra structures to incorporate sources and sinks.

2. Random Field Topic Model

Figure 2 (A) is the graphical representation of the RFT model and Figure 2 (B) shows an illustrative example. Without loss of generality we use the notations of topic modeling in language processing. A tracklet is treated as a document, and observations (points) on tracklets are quantized into words according to a codebook based on their locations and velocity directions. This analogy was used in previous work [20]. It is assumed that the spatial extents of sources and sinks of the scene are known *a priori*. An observation on a tracklet has four variables (x, z, h, m). x is the observed visual word. h and m are the labels of the source and the sink associated with the observation. If the tracklet of the observation starts from a source region or ter-

minates at a sink region, its h or m is observed. Otherwise, they need to be inferred. z is a hidden variable indicating the topic assigned to x. Λ denotes the MRF connection of neighboring tracklets. The distribution of document i over topics is specified by θ_i. $(\phi_k, \psi_k, \omega_k)$ are the model parameters of topic k. A topic corresponds to a semantic region, whose spatial distribution is speficied by ϕ_k and whose distributions over sources and sinks are specified by ψ_k and ω_k. α, β, η and κ are hyper-parameters for Dirichlet distributions. The joint distribution is

$$\begin{aligned} &p(\{(x_{in}, z_{in}, h_{in}, m_{in})\}, \{\theta_i\}, \{(\phi_k, \psi_k, \omega_k)\} | \alpha, \beta, \eta, \kappa) \\ =&\prod_k p(\phi_k|\beta) p(\psi_k|\eta) p(\omega_k|\kappa) \prod_i p(\theta_i|\alpha) \\ &p(\{z_{in}\}|\{\theta_i\}) \prod_{i,n} p(x_{in}|\phi_{z_{in}}) p(h_{in}|\psi_{z_{in}}) p(m_{in}|\omega_{z_{in}}). \end{aligned} \tag{1}$$

i, n and k are indices of documents, words and topics. θ_i, ϕ_k, ψ_k and ω_k are multinomial variables sampled from Dirichlet distributions, $p(\phi_k|\beta)$, $p(\psi_k|\eta)$, $p(\omega_k|\kappa)$ and $p(\theta_i|\alpha)$. x_{in}, h_{in} and m_{in} are discrete variables sampled from discrete distributions $p(x_{in}|\phi_{z_{in}})$, $p(h_{in}|\psi_{z_{in}})$ and $p(m_{in}|\omega_{z_{in}})$. $p(\{z_{in}\}|\{\theta_i\})$ is specified by MRF,

$$p(\mathbf{Z}|\theta) \propto \exp\left(\sum_i \log\theta_i + \sum_{j \in \varepsilon(i)} \sum_{n_1, n_2} \Lambda(z_{in_1}, z_{jn_2})\right). \tag{2}$$

$\mathbf{Z} = \{z_{ij}\}$ and $\theta = \{\theta_i\}$. $\varepsilon(i)$ is the set of tracklets which have dependency with tracklet i and it is defined by the structure of MRF. Λ weights the dependency between tracklets. Two types of MRF are defined in the following sections. The intuition behind our model is interpreted as follows. According to the property of topic models, words often co-occurring in the same documents will be grouped into one topic. Therefore, if two locations are connected by many tracklets, they tend to be grouped into the same semantic region. The MRF term Λ encourages tracklets which are spatially and temporally close to have similar distributions over semantic regions. Each semantic region has its preferred source and sink. Our model encourages the tracklets to have the same sources and sinks as their semantic regions. Therefore the learned spatial distribution of a semantic region will connect its source and sink regions.

2.1. Pairwise MRF

For pairwise MRF, $\varepsilon()$ is defined as pairwise neighborhood. A tracklet i starts at time t_i^s and ends at time t_i^e. Its starting and ending points are at locations (x_i^s, y_i^s) and (x_i^e, y_i^e) with velocities $\mathbf{v}_i^s = (v_{ix}^s, v_{iy}^s)$ and $\mathbf{v}_i^e = (v_{ix}^e, v_{iy}^e)$. Tracklet j is the neighbor of i ($j \in \varepsilon(i)$), if it satisfies

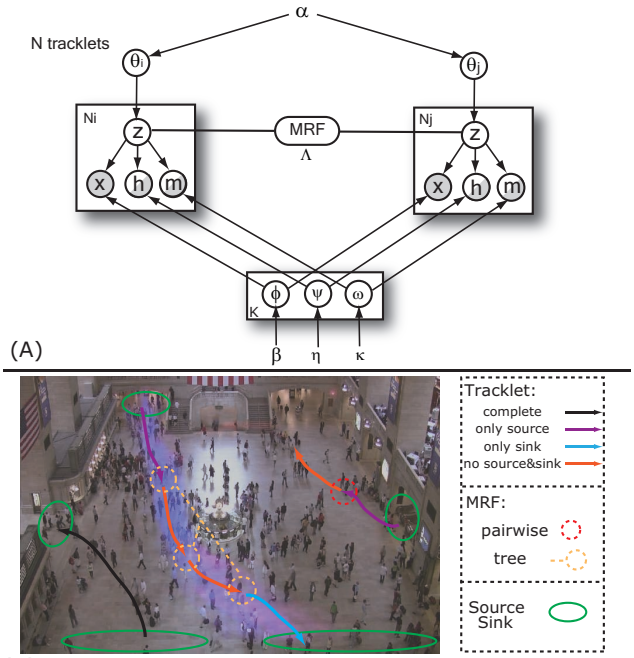

Figure 2. (A) Graphical representation of the RFT model. x is shadowed since it is observed. h and m are half-shadowed because only some of the observations have observed h and m. (B) Illustrative example of our RFT model. Two kinds of MRF connect different tracklets with observed and unobserved source/sink label to enforce their spatial and temporal coherence. The semantic region for the spanning tree is also plotted.

$$\begin{aligned} &\text{I. } t_i^e < t_j^s < t_i^e + T, \\ &\text{II. } |x_i^e - x_j^s| + |y_i^e - y_j^s| < S, \\ &\text{III. } \frac{\mathbf{v}_i^e \cdot \mathbf{v}_j^s}{\|\mathbf{v}_i^e\| \|\mathbf{v}_j^s\|} > C. \end{aligned} \tag{3}$$

I–III requires that tracklets i and j are temporally and spatially close and have consistent moving directions. We try to find pairs of tracklets which could be the same object and define them as neighbors in MRF. According to I, tracklets with temporal overlap are not considered as neighbors, since it is impossible for them to be the same objects. If these conditions are satisfied and $z_{in_1} = z_{jn_2}$,

$$\Lambda(z_{in_1}, z_{jn_2}) = \exp\left(\frac{\mathbf{v}_i^e \cdot \mathbf{v}_j^s}{\|\mathbf{v}_i^e\| \|\mathbf{v}_j^s\|} - 1\right). \tag{4}$$

Otherwise, $\Lambda(z_{in_1}, z_{jn_2}) = 0$.

2.2. Forest of randomly spanning trees

The pairwise MRF only captures the connection between two neighboring tracklets. To capture the higher-level dependencies among tracklets, the forest of randomly spanning trees is constructed on top of the neighborhood defined

Algorithm **Forest of Spanning Trees Construction**

INPUT: tracklet set $\mathcal{I}$
OUTPUT: Randomly spanning forest set $\mathcal{T}$.
01: **for** each tracklet $i \in \mathcal{I}$ **do**
02: initialize $\gamma = \emptyset$ /* γ *is one spanning tree**/
03: **Seek-tree**(i) /**Recursively search appropriate tree**/
04: **end**

function Seek-tree(tracklet m)
/* *Recursive search on neighboring tracklets defined by* Eq (3) */.
01: $\gamma \leftarrow m$
02: **if** tracklets in γ have at least one observed source **h** and **m do**
03: $\mathcal{T} \leftarrow \gamma$ /**add the tree to forest set**/
04: **break Seek-tree** /**stop current search**/
05: **end**
06: **for** each $j \in \varepsilon(m)$ **do**
07: **Seek-tree**(tracklet j)
08: **end**
09: pop out γ
end

Figure 3. Algorithm of constructing the forest of randomly spanning trees.

by the pairwise MRF. Sources and sinks are also integrated in the construction process. Sources and sinks refer to the regions where objects appear and disappear in a scene. If an object is correctly tracked all the time, its trajectory has a starting point observed in a source region and an ending point observed in a sink region. However, the sources and sinks of many tracklets extracted from crowded scenes are unknown due to tracking error. Our model assumes that the boundaries of source and sink regions of the scene are roughly known either by manual input or automatic estimation [17, 14] [2]. Experiments show that accurate boundaries are not necessary. If the starting (or ending) point of a tracklet falls in a source (or sink) region, its h (or m) is observed and is the label of that region. Otherwise h (or m) is unobserved and needs to be inferred.

The algorithm of constructing the forest of randomly spanning tree γ is listed in Figure 3. A randomly spanning tree is composed of several tracklets with pairwise connections, which are defined as the same in Eq (3). The randomly spanning tree is constructed with the constraint that it starts with a tracklet whose starting point has an observed source h and ends with a tracklet whose ending point has an observed sink m. Then $\varepsilon()$ in Eq (2) is defined by the forest of randomly spanning tree γ, *i.e.* if tracklet i and j are on the same randomly spanning tree, $j \in \gamma(i)$.

[2] In our approach, source and sink regions are estimated using the Gaussian mixture model [14]. Starting and ending points of tracklets caused by tracking failures are filtered considering the distributions of accumulated motion densities within their neighborhoods [22]. It is likely for a starting (ending) point to be in a source (sink) region, if the accumulated motion density quickly drops along the opposite (same) moving direction of its tracklet. After filtering, high-density Gaussian clusters correspond to sources and sinks. Low-density Gaussian clusters correspond to tracking failures. We skip the details since this is not the focus of this paper.

2.3. Inference

We derive a collapsed Gibbs sampler to do inference. It integrates out $\{\theta, \phi, \psi, \omega\}$ and samples $\{z, h, m\}$ iteratively.

The posterior of z_{in} given other variables is

$$\begin{aligned} & p(z_{in} = k | \mathbf{X}, \mathbf{Z}_{\backslash in}, \mathbf{H}, \mathbf{M}) \\ \propto & \frac{n^{(w)}_{k,\backslash in} + \beta}{\sum_{w=1}^{W}(n^{(w)}_{k,\backslash in} + \beta)} \frac{n^{(p)}_{k,\backslash in} + \eta}{\sum_{p=1}^{P}(n^{(p)}_{k,\backslash in} + \eta)} \\ & \frac{n^{(q)}_{k,\backslash in} + \kappa}{\sum_{q=1}^{Q}(n^{(q)}_{k,\backslash in} + \kappa)} \frac{n^{(k)}_{i,\backslash n} + \alpha}{\sum_{k=1}^{K}(n^{(k)}_{i,\backslash n} + \alpha)} \\ & \exp\left(\sum_{j \in \gamma(i)} \sum_{n'} \Lambda(z_{in}, z_{jn'}) \right). \end{aligned} \tag{5}$$

$\mathbf{X} = \{x_{in}\}, \mathbf{Z} = \{z_{in}\}, \mathbf{H} = \{h_{in}\}, \mathbf{M} = \{m_{in}\}$. Subscript $\backslash in$ denotes counts over the whole data set excluding observation n on tracklet i. Denote that $x_{in} = w, h_{in} = p, m_{in} = q$. $n^{(w)}_{k,\backslash in}$ denotes the count of observations with value w and assigned to topic k. $n^{(p)}_{k,\backslash in}$ ($n^{(q)}_{k,\backslash in}$) denotes the count of observations being associated with source p (sink q) and assigned to topic k. $n^{k}_{i,\backslash n}$ denotes the count of observations assigned to topic k on tracklet i. W is the codebook size. P and Q are the numbers of sources and sinks. The posteriors of h_{in} and m_{in} given other variables are,

$$p(\mathrm{h}_{in} = p | \mathbf{X}, \mathbf{Z}, \mathbf{H}_{\backslash i}, \mathbf{M}) \propto \frac{n^{(p)}_{k,\backslash in} + \eta}{\sum_{p=1}^{P}(n^{(p)}_{k,\backslash in} + \eta)}, \tag{6}$$

$$p(\mathrm{m}_{in} = q | \mathbf{X}, \mathbf{Z}, \mathbf{H}, \mathbf{M}_{\backslash in}) \propto \frac{n^{(q)}_{k,\backslash in} + \kappa}{\sum_{q=1}^{Q}(n^{(q)}_{k,\backslash in} + \kappa)}. \tag{7}$$

If h_{in} and m_{in} are unobserved, they are sampled based on Eq (6) and (7). Otherwise, they are fixed and not updated during Gibbs sampling. After sampling converges, $\{\theta, \psi, \omega\}$ could be estimated from any sample by

$$\hat{\theta}^{(w)}_{k} = \frac{n^{(w)}_{k} + \beta}{\sum_{w=1}^{W}(n^{(w)}_{k} + \beta)}, \tag{8}$$

$$\hat{\psi}^{(p)}_{k} = \frac{n^{(p)}_{k} + \eta}{\sum_{p=1}^{P}(n^{(p)}_{k} + \eta)}, \tag{9}$$

$$\hat{\omega}^{(q)}_{k} = \frac{n^{(q)}_{k} + \kappa}{\sum_{q=1}^{Q}(n^{(q)}_{k} + \kappa)}. \tag{10}$$

Algorithm **Optimal Spanning Tree Ranking**

INPUT: the online tracklet g, the learnt tracklet set $\mathcal{I}$
OUTPUT: Optimal spanning tree $\tilde{\gamma}(g)$ and $\mathbf{z}_{\tilde{\gamma}}$ for g.
01: Exhaustively Seek neighbor grids ε of trajectory g
based on Constraint II and III in set $\mathcal{I}$
02: **for** each ε_i **do**
03: $\gamma_i \leftarrow$ **Seek-tree**(g) on ε_i
04: Gibbs Sampling for $\mathbf{z}_{\gamma_i}$
03: $\mathcal{P} \leftarrow \gamma_i$ */ * $\mathcal{P}$ is the potential tree set * /*
04: **end**
05: $\tilde{\gamma}(g)$=$\underset{\gamma \in \mathcal{P}}{\operatorname{argmin}} H(Z_\gamma)$
/ $H(Z) = -\sum_z p(z) \log p(z)$ *is the information entropy, computed over distribution of* $\mathbf{z}$ *for the spanning tree* γ_i*, to select the optimal spanning tree */.*

Figure 4. Algorithm of obtaining the optimal spanning tree for online tracklet.

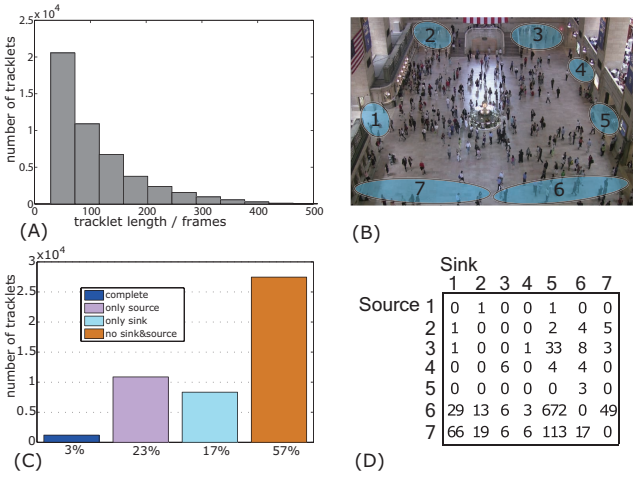

Figure 5. (A) The histogram of tracklet lengths. (B) Detected source and sink regions. (C) Statistics of sources and sinks of all the tracklets. (D) The summary of observed sources and sinks of the complete tracklets.

Once the RFT model is learnt, tracklets can be clustered based on the semantic regions they belong to. A tracklet is assigned to semantic region k if most of its points are assigned to this semantic region.

2.4. Online tracklet prediction

After semantic regions are learned, our model can online analyze the tracklets, *i.e.* classifying them into semantic regions and predicting their sources and sinks. It is unreliable to analyze an online tracklet alone using the models of semantic regions, since when the tracklet is short it may fit more than one semantic region. Instead, we first obtain its optimal spanning tree from the training set using the algorithm in Figure 4. It is assumed that a pedestrian's behavior at one location is statistically correlated to the behaviors of pedestrians in the training set at the same location. The algorithm first correlates the online tracklet with the tracklets from the training set by generating several spanning trees. The spanning tree with the minimum entropy on $\mathbf{z}$ is chosen for the online tracklet to infer its topic label, source, and sink.

3. Experiments

Experiments are conducted on a 30 minutes long video sequence collected from the New York's Grand Central station. Figure 2 (B) shows a single frame of this scene. The video is at the resolution of 480×720. $47,866$ tracklets are extracted. The codebook of observations is designed as follows: the 480×720 scene is divided into cells of size 10×10 and the velocities of keypoints are quantized into four directions. Thus the size of the codebook is $48 \times 72 \times 4$.

Figure 5 shows the summary of collected tracklets. (A) is the histogram of tracklet lengths. Most of tracklet lengths are shorter than 100 frames. (B) shows the detected sources and sinks regions indexed by $1 \backsim 7$. (C) shows the percentages of four kinds of tracklets. Only a very small portion of tracklets (3%) (labeled as "*complete*") have both observed sources and sinks. 24% tracklets (labeled as "*only source*") only have observed sources. 17% tracklets (labeled as "*only sink*") only have observed sinks. For more than half of tracklets (56%), neither sources nor sinks are observed. (D) summarizes the observed sources and sinks of the complete tracklets. The vertical axis is the source index, and horizontal axis is the sink index. It shows that most complete tracklets are between the source/sink regions 5 and 6 since they are close in space. Therefore, if only complete tracklets are used, most semantic regions cannot be well learned. Note that all tracklets come directly from the KLT tracker, no preprocessing is involved in correcting the camera distortion for tracklets.

Hyper-parameters $\alpha, \beta, \eta, \kappa$ are uniform Dirichlet distributions and are empirically chosen as 1. Our results are not sensitive to these parameters. They serve as priors of Dirichlet distributions to avoid singularity of the model, the general discussion for the influence of the hyper-parameters on learning topic model could be found in [2]. It takes around 2 hours for the Gibbs sampler to converge on this data set, running on a computer with 3GHz core duo CPU in Visual C++ implementation. The convergence is empirically determined by the convergence of data likelihood, when the variation of data likelihood becomes trivial after hundreds of iteration of Gibbs sampling. The online tracklet prediction takes 0.5 seconds per tracklet.

3.1. Learning semantic regions

Our RFT model using the forest of randomly spanning trees learns 30 semantic regions in this scene. In the learning process, around 23,000 randomly spanning trees are constructed, and one tracklet may belong to more than one randomly spanning tree. Figure 6 (A) visualizes some representative semantic regions[3]. According to the learned $\hat{\psi}$ and $\hat{\omega}$, the most probable source and sink for each semantic region are also shown. The learned semantic regions represent the primary visual flows and paths in the scene. They spatially expand in long ranges and well capture the global structures of the scene. Meanwhile, most paths are well separated and many structures are revealed at fine scales with reasonably good accuracy. Most learned semantic regions only have one source and one sink, except semantic region 19 which has two sources. Semantic region 14 also diverges. The results of these two regions need to be improved. It is observed that sources and sinks, whose boundaries are defined beforehand, only partially overlap with their semantic regions. One source or sink may correspond to multiple semantic regions. This means that although the prior provided by sources and sinks effectively guides the learning of semantic regions, it does not add strong regularization on the exact shapes of semantic regions. Therefore our model only needs the boundaries of sources and sinks to be roughly defined.

For comparison, the results of optical flow based HDP (OptHDP) model [21] and trajectory based Dual HDP (TrajHDP) [20] are shown in Figure 6 (B) and (C). Both methods are based on topic models. OptHDP learns the semantic regions from the temporal co-occurrence of optical flow features and it was reported to work well in traffic scenes [21]. It assumed that at different time different subsets of activities happened. If two types of activities always happen at the same time, they cannot be distinguished. In our scene, pedestrians move slowly in a large hall. For most of the time activities on different paths are simultaneously observed with large temporal overlaps. Temporal co-occurrence information is not discriminative enough in this scenario. As a result, different paths are incorrectly merged into one semantic region by OptHDP as shown in Figure 6 (B). TrajHDP is related to our method. It assumed that a significant portion of trajectories were complete and that if two locations were on the same semantic region they were connected by many trajectories. However, a large number of complete trajectories are unavailable from this crowded scene. Without MRF and source-sink priors, TrajHDP can only learn semantic regions expanded in short ranges. Some paths close in space are incorrectly merged. For example, the two paths (21 and 15 in Figure 6 (A)) learned by our approach are close in the bottom-right region of the scene. They are separated by our approach because they diverge toward different sinks in the top region. However, since TrajHDP cannot well capture long-range distributions, they merge into one semantic region shown in the fifth row of Figure 6 (C). Overall, the semantic regions learned by our approach are more accurate and informative than OptHDP and TrajHDP.

[3]The complete results of semantic regions and tracklet clustering can be found in our supplementary material.

3.2. Tracklet clustering based on semantic regions

Figure 7 (A) shows some representative clusters of tracklets obtained by our model using the forest of randomly spanning trees as MRF prior. Even though most tracklets are broken, some tracklets far away in space are also grouped into one cluster because they have the same semantic interpretation. For example, the first cluster shown in Figure 7 (A) contains tracklets related to the activities of "pedestrians from source 2 walk toward sink 7". It is not easy to obtain such a cluster, because most tracklets in this cluster are not observed either in source 2 or in sink 7. Figure 7 (B) and (C) show the representative clusters obtained by Hausdorff distance-based Spectral Clustering (referred as SC) [1] and TrajHDP [20]. They are all in short range spatially and it is hard to interpret their semantic meanings.

To further quantitatively evaluate the clustering performance, we use *correctness* and *completeness* introduced in [15] as measurements of the clustering accuracy. Correctness is the accuracy that two tracklets, which belong to different activity categories based on the ground truth, are also grouped into different clusters by the algorithm. Completeness is the accuracy that two tracklets, which belong to the same activity category, are also grouped into the same cluster by the algorithm. In extreme cases, if all the tracklets are grouped into one cluster, the completeness is 100% while the correctness is 0%; if every tracklet is put into a different cluster, the completeness is 0% while the correctness is 100%. A good cluster algorithm should have both high correctness and high completeness. To measure correctness (completeness), we manually label 2000 (1507) pairs of tracklets and each pair of tracklets belong to different (the same) activity categories (category) as ground truth. The accuracies of correctness and completeness for our pairwise RFT model, tree RFT model, TrajHDP [20] and SC [1] are reported in Table 1. Our tree RFT model achieves the best performance in terms of both correctness and completeness. The pairwise RFT model also outperforms TrajHDP and SC. Note that the correctness is low when the cluster number is 2, since many trajectories of different paths have to be put into one cluster. The completeness is low when the cluster number is large, since trajectories of the same path are divided into different clusters.

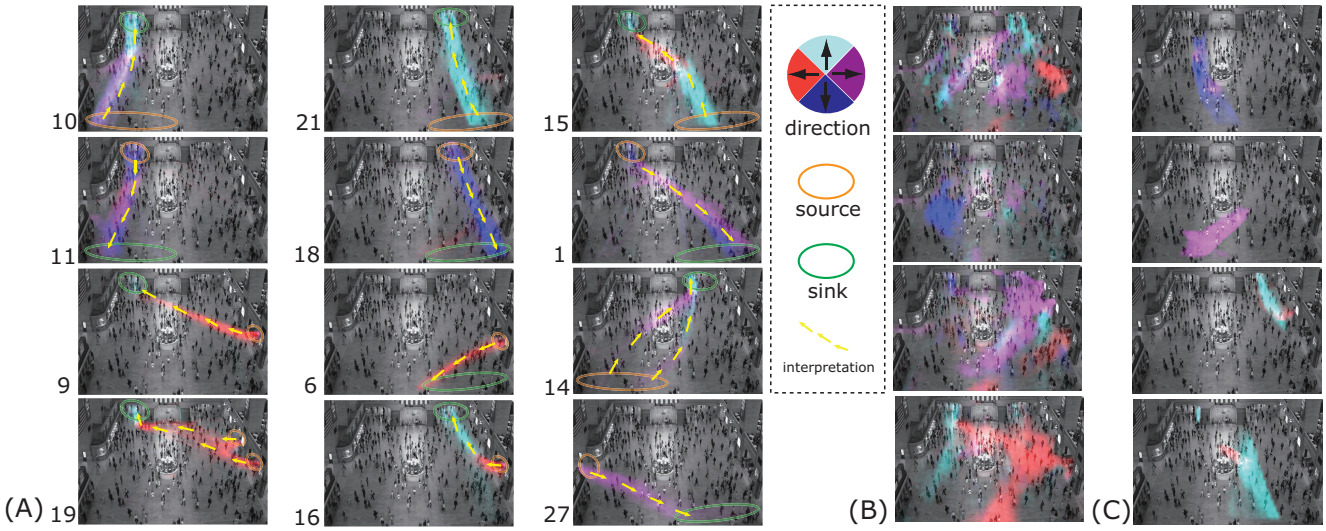

Figure 6. Representative semantic regions learned by (A) our model (semantic region indices are randomly assigned by learning process), (B) OptHDP [21] and (C) TrajHDP [20]. The velocities are quantized into four directions represented by four colors. The two circles on every semantic region represent the learned most probable source and sink. The boundaries of sources and sinks in the scene are pre-detected and shown in Figure 5 (A). (Better view in color version)

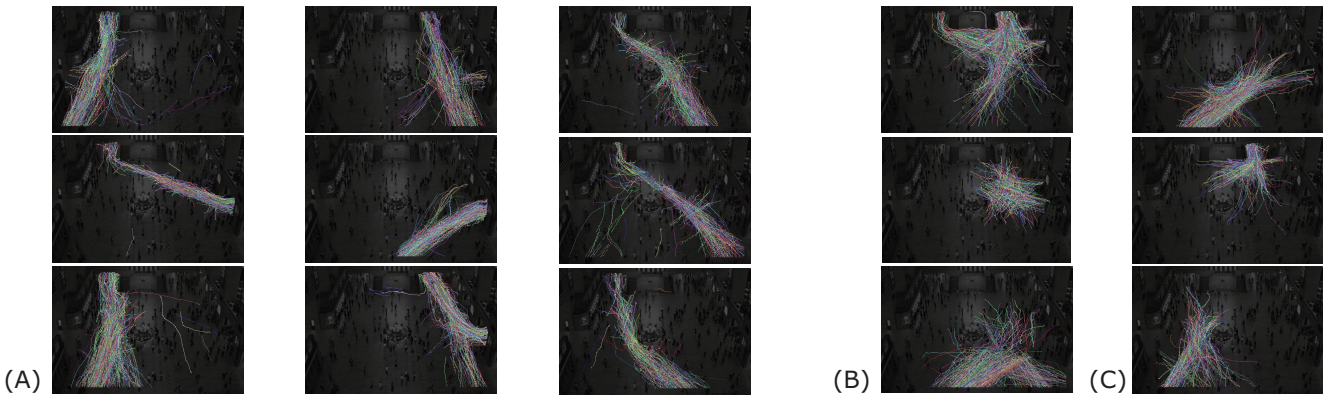

Figure 7. Representative clusters of trajectories by (A)our model, (B)SC [1] and (C)TrajHDP [20]. Colors of every trajectories are randomly assigned.

4. Discussion and Conclusion

In this paper we proposed a new approach of learning semantic regions of crowded scenes from tracklets, which are a mid-level representation between local motions and complete trajectories of objects. It effectively uses the MRF prior to capture the spatial and temporal dependency between tracklets and uses the source-sink prior to guide the learning of semantic regions. The learned semantic regions well capture the global structures of the scenes in long range with clear semantic interpretation. They are also able to separate different paths at fine scales with good accuracy. Both qualitative and quantitative experimental evaluations show that it outperforms state-of-the-art methods.

Our model also has other potential applications to be explored. For example, after inferring the sources and sinks of tracklets, the transition probabilities between sources and sinks can be estimated. It is of interest for crowd control and flow prediction. Figure 8(A)(B) show the transition probabilities from sources 2 and 6 to other sinks learned by our RFT model. Our model can also predict the past and future behaviors of individuals whose existence is only partially observed in a crowded scene. As shown in Figure 8(C)(D), two individuals are being tracked, two online tracklets are generated. With the algorithm in Figure 4 to obtain the optimal spanning tree, our model could predict the most possible compact paths of the individuals and estimate where they came from and where they would go. To estimate individual behavior in public crowded scenes is a critical feat

Cluster Number		2	5	8	11	14	17	20	23	26	29	32
Our Tree RFT	Completeness	**0.93**	**0.79**	**0.75**	**0.79**	**0.74**	**0.71**	**0.66**	**0.67**	**0.65**	**0.61**	**0.61**
	Correctness	0.47	**0.81**	**0.89**	**0.92**	**0.94**	**0.95**	**0.97**	**0.97**	**0.97**	**0.98**	**0.98**
Our Pairwise RFT	Completeness	0.77	0.65	0.62	0.62	0.58	0.62	0.58	0.55	0.60	0.57	0.54
	Correctness	**0.49**	0.78	0.86	0.88	0.89	0.92	0.93	0.94	0.95	0.96	0.95
SC	Completeness	0.60	0.39	0.26	0.19	0.18	0.15	0.12	0.11	0.11	0.08	0.08
	Correctness	**0.49**	**0.81**	0.88	0.90	0.92	0.94	0.95	0.95	0.96	0.96	0.96
TrajHDP	Completeness	0.35(cluster number is 25)										
	Correctness	0.92(cluster number is 25)										

Table 1. Completeness and correctness of our Tree-based RFT and Pairwise RFT models, and Spectral Clustering(SC) [7], with respect to cluster numbers. The completeness and correctness of TrajHDP[20] are also shown and it finds cluster number automatically as 25.

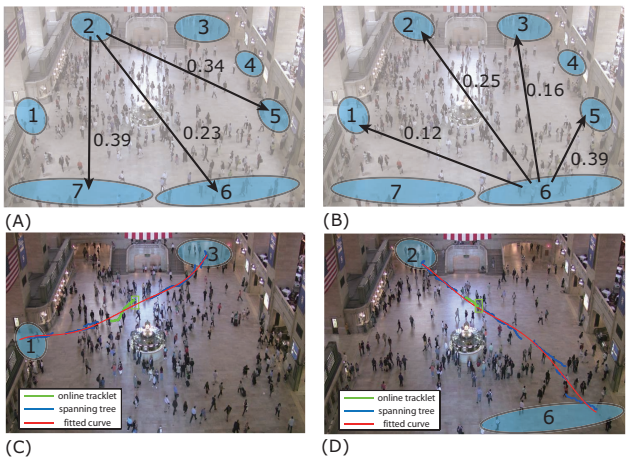

Figure 8. (A)(B) Transition probabilities from sources 2 and 6 to other sinks. Only some major transition modes are shown. (C)(D) Two online tracklets are extracted, and their optimal spanning trees are obtained. The fitted curve for spanning trees predict the compact paths of the individuals and their most possible entry and exit locations are also estimated by our RFT model.

for intelligent surveillance systems. These applications will be explored in details in the future work.

5. Acknowledgement

This work is supported by the General Research Fund sponsored by the Research Grants Council of Hong Kong (project No. CUHK417110) and National Natural Science Foundation of China (project no. 61005057).

References

[1] S. Atev, O. Masoud, and N. Papanikolopoulos. Learning traffic patterns at intersections by spectral clustering of motion trajectories. In *Proc. IEEE Conf. Intell. Robots and Systems*, 2006.

[2] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 2003.

[3] W. Ge and R. Collins. Multi-target data association by tracklets with unsupervised parameter estimation. In *Proc. BMVC*, 2008.

[4] T. Hospedales, S. Gong, and T. Xiang. A markov clustering topic model for mining behaviour in video. In *Proc. ICCV*, 2009.

[5] J. W. Hsieh, Y. S. H., Y. S. Chen, and W. Hu. Automatic traffic surveillance system for vehicle tracking and classification. *IEEE Trans. on Intelligent Transportation Systems*, 2006.

[6] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. Maybank. A system for learning statistical motion patterns. *IEEE Trans. on PAMI*, 2006.

[7] W. Hu, D. Xie, Z. Fu, W. Zeng, and S. Mayband. Semantic-based surveillance video retrieval. *IEEE Trans. on Image Processing*, 2007.

[8] E. Keogh and M. Pazzani. Scaling up dynamic time warping for datamining applications. In *Proc. ACM SIGKDD*, 2000.

[9] J. Li, S. Gong, and T. Xiang. Global behavior inference using probabilistic latent semantic analysis. In *Proc. BMVC*, 2008.

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

[11] J. Li, S. Gong, and T. Xiang. Discovering multi-camera behaviour correlations for on-the-fly global activity prediction and anomaly detection. In *Proc. of IEEE Int'l Workshop on Visual Surveillance*, 2009.

[12] X. Liu, L. Lin, S. Zhu, and H. Jin. Trajectory parsing by cluster sampling in spatio-temporal graph. In *Proc.CVPR*, 2009.

[13] C. Loy, S. Gong, and T. Xiang. Multi-camera activity correlation analysis. In *Proc.CVPR*, 2009.

[14] D. Makris and T. Ellis. Learning semantic scene models from observing activity in visual surveillance. *IEEE Trans. on SMC*, 2005.

[15] B. Moberts, A. Vilanova, and J. van Wijk. Evaluation of fiber clustering methods for diffusion tensor imaging. In *IEEE Visualization*, 2005.

[16] I. Saleemi, K. Shafique, and M. Shah. Probabilistic modeling of scene dynamics for applications in visual surveillance. *IEEE Trans. on PAMI*, 2009.

[17] C. Stauffer. Estimating tracking sources and sinks. In *Computer Vision and Pattern Recognition Workshop*, 2003.

[18] C. Tomasi and T. Kanade. Detection and tracking of point features. *Int'l Journal of Computer Vision*, 1991.

[19] J. Verbeek and B. Triggs. Region classification with markov field aspect models. In *Proc.CVPR*, 2007.

[20] X. Wang, K. Ma, G. Ng, and W. Grimson. Trajectory analysis and semantic region modeling using a nonparametric bayesian model. In *Proc.CVPR*, 2008.

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

[22] X. Wang, K. Tieu, and E. Grimson. Learning semantic scene models by trajectory analysis. *Proc. ECCV*, 2006.

[23] X. Wang, K. Tieu, and W. Grimson. Correspondence-free activity analysis and scene modeling in multiple camera views. *IEEE Trans. on PAMI*, 2010.

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

[25] T. Zhang, H. Lu, and S. Z. Li. Learning semantic scene models by object classification and trajectory clustering. In *Proc.CVPR*, 2009.