2014 22nd International Conference on Pattern Recognition

# A Belief based Correlated Topic Model for Trajectory Clustering in Crowded Video Scenes

Jialing Zou[1], Qixiang Ye[1,2], Yanting Cui[1], David Doermann[2], *Fellow, IEEE*, Jianbin Jiao[1*]
[1]University of Chinese Academy of Sciences, Beijing 101408, China.
[2]Institute for Advanced Computer Studies, University of Maryland, College Park
[2]{qxye, doermann}@umiacs.umd.edu, [1*]jiaojb@ucas.ac.cn

***Abstract*—Trajectory clustering in crowded video scenes is very challenging. In this paper, we propose to use a belief based correlated topic model (BCTM) to learn discriminative middle level features for trajectory clustering. By constructing a scene prior based joint Gaussian distribution, the BCTM can uncover relations between trajectory clusters and the middle level features using a parameter estimation procedure. The method has distinct advantages over Correlated Topic Model (CTM) and Random Field Topic (RFT) model previously proposed. The inputs to the BCTM are either full trajectories or trajectory fragments obtained with an existing tracking algorithm. The output BCTM features are input to a hierarchical clustering algorithm to obtain trajectory clusters. Experiments on three benchmark datasets show that the proposed BCTM and trajectory clustering approach improves the state of the art.**

## I. Introduction

Trajectory clustering is a video analysis task whose goal is to assign individual trajectories with common cluster labels, with applications in activity surveillance, traffic flow estimation and emergency response [1], [2].

A straightforward way to do trajectory clustering is to use low level feature classification. A set of features, i.e., coordinates, velocities and/or geometrical shapes, are extracted to represent trajectories, and then unsupervised learning methods or inference methods are used to classify these features [1], [3], [4]. However, in crowded video scenes, it is often difficult to obtain complete trajectories with off-the-shelf tracking methods [5]. In most cases, incomplete short trajectories with noise are obtained, which are difficult to use in low level feature based trajectory clustering and analysis.

In recent years, middle level feature based trajectory analysis and clustering approaches have attracted attentions. In surveillance videos of crowded scenes, middle level features are usually observed as dominant paths of objects, which provide a reasonable representation of incomplete trajectories in low dimensional feature space [6]. With these middle level features, the cluster information of trajectories is more intuitive, and the correspondence between the features and their clusters can be better modeled.

Middle level features for trajectory clustering can be learned with non-Bayesian approaches, for example, similarity clustering [7], dimension reduction [8] or online cluster update [9]. In [7], Wang et al. defined two Euclidean similarity measures, and trajectories are clustered on the defined measures. In [8], Hu et al. introduced a dimensional spectral clustering method, trajectories are projected to a lower space through eigenvalue factorization, and clustered in the lower sub-space

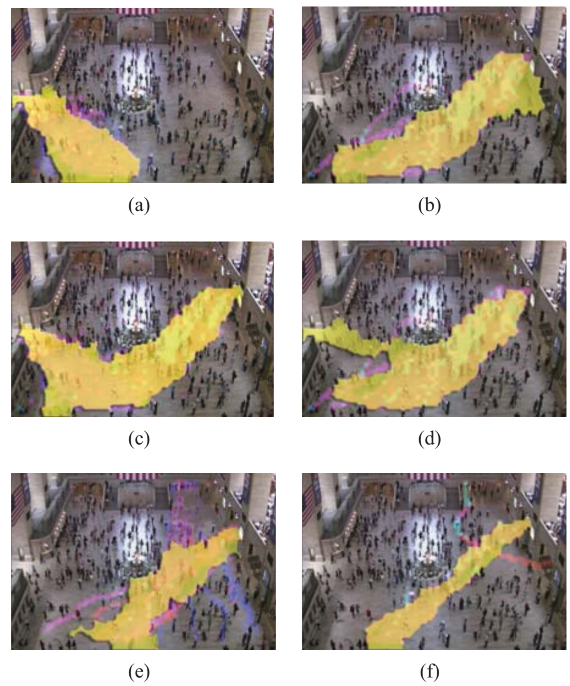


Fig. 1. (a) and (b) are two trajectory clusters. (c), (d), (e), (f) are four learned topics. It can be seen that the topics are not consistent with clusters. Topics can be shared with clusters, and clusters can also be shared with topics.

with a $k$-means algorithm. In [9], Hu et al. proposed an online cluster updating approach with a fuzzy $k$-means for online trajectory clustering. Despite the simplicity of the above approaches, however, they often require long and complete trajectories as input. In addition, it is difficult to include scene priors with clustering models.

Recently, hierarchical latent variable Bayesian models, such as latent Dirichlet allocation (LDA) [10] and hierarchical Dirichlet process (HDP) [11], have been widely explored in trajectory analysis and clustering [2], [12], [13], [14]. These models were adopted from the literature classification and are well known as "topic models". They often have hierarchical structures, and the latent variables tie the different levels. With LDA or HDP models, trajectories are treated as documents and object observations of trajectories are treated as visual words. In our case, learned topics correspond to the middle level features of trajectories.

In [13], Wang et al. proposed to perform trajectory clus-

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.439



IEEE computer society

tering using a mixture of latent Dirichlet allocation (LDA) models, which enables trajectories to share different Dirichlet distributions. In [12], [14], Wang et al. proposed a dual hierarchical Dirichlet process (Dual-HDP) model and a dynamic dual hierarchical Dirichlet process (DDual-HDP) for trajectory clustering. Dual-HDP would co-cluster the words and documents with automatically decided topics. The DDual-HDP model is the extension of Dual-HDP to incorporate online learning. In [2], Zhou et al. proposed a Random Field Topic (RFT) model to perform trajectory clustering in crowded video scenes. The RFT model advances the LDA and HDP models, by integrating belief priors and using a Markov random field (MRF) based minimum spanning tree algorithm. In Zhou's approach, the learned topics (middle level features) are called semantic regions. The performance of trajectory clustering is significantly improved over LDA and HDP models, however, problems remain. Existing approaches usually depend on the topic information, and can not completely regularize topics for clusters, even with belief priors. This can drop the clustering performance in crowded scenes, where trajectory topics could be correlated. This occurs, for example, when topics are shared with different clusters, and clusters are also shared with different topics, as illustrated in Fig. 1.

To model the correlated topics and clusters, a correlated topic model (CTM) [15] is proposed to eliminate the independent assumption of topics with a Gaussian distribution. In [16], Rodriguez et al. proposed to use CTM to extract middle level features and perform large scale crowd behavior analysis. However, as CTM can not create discriminative middle level features for different clusters, these approaches often require a combination of the middle level features with some low level features.

In this paper, a scene prior belief based correlated topic model (BCTM) is proposed to learn discriminative middle level features (topics) for trajectory clustering. The use of prior belief is based on the observation that most moving objects have a clear indication about where they come from (sources) and where they want to go (sinks). Beliefs are set as sources and sinks of objects. Considering that even with the beliefs, the trajectory clusters are often not completely consistent with the topics [17], we propose to construct a scene prior based joint Gaussian distribution over the topics, and use the covariance to reflect the relations between topics and clusters.

For trajectory clustering, a KLT tracker [18] is employed to get "trajectory fragments". A Markov random field (MRF) based spanning tree [2] is used to link the fragments into trees to capture initial cluster information. The middle level features are then extracted by the proposed BCTM. On the learned middle level features, we use a hierarchical clustering algorithm to get trajectory clusters.

## II. Belief Based Correlated Topic Model

This section presents the BCTM, and processes the trajectory middle level feature extraction with BCTM parameter estimation.

### A. Correlated Topic Model

To make the paper self-contained, we first review the CTM. Fig. 2(a) shows the graphical representation of CTM [15]. The four important notations about CTM are corpus, document, topic and word, which correspond to path, trajectory, topic and visual word. For simplicity, we omit subscripts of parameters and variables in Fig. 2. $M$ denotes the number of documents. $\eta$ is assumed to follow a joint Gaussian distribution $\mathcal{N}(\mu, \Sigma)$. $N$ denotes the number of words for each document. $z$ is a latent variable being assumed to follow a parameterized multinomial distribution $Mult(f(\eta))$. The function $f$ discretizes the continuous variable $\eta$ so that $f(\eta)$ can be a valid parameter in the multinomial distribution. $x$ denotes words and $\beta$ denotes hyper-paramerers.

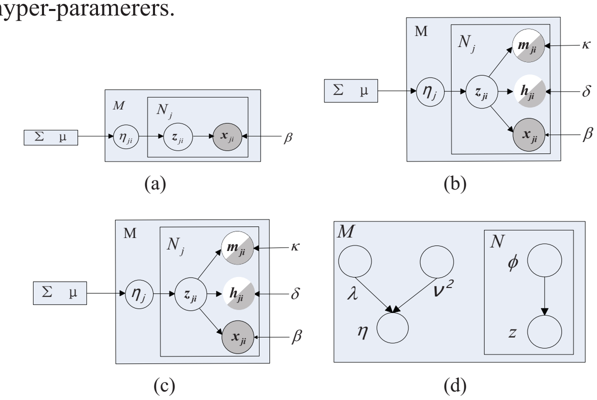



Fig. 2. (a) and (b) are the graphical representations of CTM and BCTM without subscripts. (c) and (d) are the graphical representations of BCTM and approximate graph model of BCTM with subscripts.

### B. Belief Correlated Topic Model

In Fig. 2(b), it can be seen that words with partially observed beliefs are inputs for the BCTM. $h$ and $m$ are binary $s$ dimensional vectors, and $s$ is the number of beliefs. For convenience and efficiency, we create the visual words with tracked coordinates and velocities of objects. The detail of the creation of visual words is described in Section III. In Fig. 2(b), it can be seen that the observation of a visual word has four variables ($x$, $h$, $m$, $z$), where $x$ is a visual word, $h$ and $m$ are the labels of beliefs of $x$, and $z$ is a latent variable indicating $x$'s topics. $h$ or $m$ are observed if the trajectories start or end within the annotated belief areas in Fig. 4. Otherwise, they are inferred. The distribution of a trajectory is specified by $\eta$. $\beta$, $\delta$ and $\kappa$ are hyper-parameters for multinomial distributions of words and beliefs, respectively. According to Fig. 2(c), which is the BCTM graph representation with subscripts, the joint distribution is

$$\begin{aligned} & p(\{(x_{ji}, h_{ji}, m_{ji}, z_{ji})\}, \{\eta_j\}|\Sigma, \mu, \beta, \delta, \kappa) \\ = & \prod_j p(\eta_j|\Sigma, \mu)[\prod_{j,i} p(z_{ji}|f(\eta_j))p(x_{ji}|\beta_{z_{ji}}) \\ & p(h_{ji}|\delta_{z_{ji}})p(m_{ji}|\kappa_{z_{ji}})], \end{aligned} \tag{1}$$

where $j$, $i$ are indices of documents (trajectories) and visual words, $\eta_j$ is a continuous variable sampled from a multivariate Gaussian distribution $p(\eta_j|\mu, \Sigma)$, and $x_{ji}$, $h_{ji}$, $m_{ji}$ are discrete variables sampled from discrete distributions $p\left(x_{ji}|\beta_{z_{ji}}\right)$, $p\left(h_{ji}|\delta_{z_{ji}}\right)$, $p\left(m_{ji}|\kappa_{z_{ji}}\right)$, respectively.

### C. Parameter Estimation for Middle Level Feature Extraction

Middle level features are extracted through parameter estimation using BCTM. The common way to estimate parameters in a graph model is to iteratively maximize the posterior likelihood. The *log* posterior likelihood of BCTM is given by

$$\begin{aligned} &\log[p(x_{ji}, h_{ji}, m_{ji}|\Sigma, \mu, \beta, \delta, \kappa)] \\ &= \log\{\int [\sum_{z_{ji}} p(\eta_j, z_{ji}, x_{ji}, h_{ji}, m_{ji}|\Sigma, \mu, \beta, \delta, \kappa)] d_{\eta_j}\}. \end{aligned} \quad (2)$$

However, the *log* posterior likelihood in (2) is intractable [15]. A variational breaking method [19] is employed to do the approximation with a graph model shown in Fig. 2(d). In the approximate graph model, the conditional dependence between $x_{ji}$ and $h_{ji}$, and $\eta_j$ and $z_{ji}$ are eliminated. $z_{ji}$ is conditioned on a new variable $\phi_{ki}$, which is independent of $\eta_j$. $\lambda$ and $\nu^2$ are the means and covariance of the joint Gaussian distribution. The subscript $k$ denotes the number of topics, and each document has a $\phi$. The gap between the true and approximate *log* posterior likelihood is the KL divergence. The approximate is shown in (3).

$$\begin{aligned} L = &E_q[\log p(\eta_j|\Sigma, \mu)] \\ &+E_q(\log p(z_{ji}|\eta_j)) + E_q[\log p(x_{ji}|z_{ji}, \beta)] \\ &+E_q[\log p(h_{ji}|z_{ji}, \delta)] + E_q[\log p(m_{ji}|z_{ji}, \kappa)] \\ &-E_q[\log q(\eta_j)] - E_q[\log q(z_{ji})]. \end{aligned} \quad (3)$$

We iteratively maximize L(•) by computing the derivatives of L(•) with respect to different variables and parameters. This step iteratively minimizes the KL divergence between the true and approximated posterior likelihood. For details of computation, please refer to [15]. We give modified parameters and variables as

$$\phi_{ki} \propto \exp(\lambda_j) * \beta_{ki} * \delta_{ki} * \kappa_{ki}, \quad (4)$$

$$\beta_k \propto \sum_i \phi_{ki} n_i^x, \quad (5)$$

$$\kappa_k \propto \sum_i \phi_{ki} n_i^h, \quad (6)$$

$$\delta_k \propto \sum_i \phi_{ki} n_i^m, \quad (7)$$

where $n_i^x$, $n_i^h$ and $n_i^m$ denote the word count, belief (source and sink) count, respectively. $\phi_{ki}$ denotes the $i$th word $k$ topic probability. $\beta_k$, $\kappa_k$ and $\delta_k$ denote the $k$th topic (middle level feature) representation in the word space, belief (source and sink) space, respectively.

It can be seen that $\beta$ contains the middle level features. According to (5), it is proportional to $\phi$ and $n^x$, and $\phi$ is proportional to $\lambda$, $\beta$, $\delta$ and $\kappa$. That means the words with different beliefs have different $\phi$. Therefore, the middle level features $\beta$ learned by BCTM are more discriminative than those learned by CTM, where $\phi$ is only proportional to $\lambda$ and $\beta$.

It is noted that the mean parameters $\lambda$ will be iteratively optimized by the covariance $\Sigma$ of the Gaussian distribution. It is not enough however to indirectly utilize the covariance for trajectory clustering. CTM can also construct the covariance with a Gaussian distribution, but without discriminative middle level features (topics), the covariance can not reflect the right relations between clusters and middle level features. The LDA based RFT model [2] integrates the belief to learn discriminative middle level features, however it also assumes that the middle level features are independent to each other. In contrast, our proposed BCTM can not only construct the covariance among the middle level features, but also get discriminative features. It is important to the following hierarchical clustering algorithm.

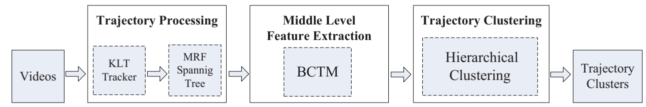



Fig. 3. Flowchart of trajectory clustering.

## III. Hierarchical Trajectory Clustering

In this section we describe how the trajectories in crowded videos are clustered, as shown in the flowchart of Fig. 3.

As discussed above, the topic labels are not consistent with the cluster labels. Therefore, there are over-segmented and under-segmented topics (middle level features) after the model estimation and inference procedure. An over-segmented topic implies that the topic shares a cluster with other topics and an under-segmented topic implies that the topic is shared with multiple clusters. A topic could be simultaneously under-segmented and over-segmented with respect to different clusters. In the hierarchical clustering algorithm, the objective is to merge multiple over-segmented topics into one cluster, as well as split each under-segmented topic into multiple clusters.

We first use a KLT tracker [18] to calculate trajectory segments and motion vectors of objects, and automatically identify the belief areas as shown in Fig. 4, referring to [20]. In order to get belief labels and the initial clusters, trajectory segments are converted into trajectory trees by a spanning tree algorithm [2]. The trees are then used to extracted visual words, which are input to the BCTM for middle level feature extraction. In the middle level feature extraction procedure, topic probabilities of trees in the topic space are computed. Given $K$ topics, each tree has $K$ corresponding probabilities. The topic label of the largest probability is assigned to the tree. After this step, each trajectory segment has an initial topic label. The topics of initial labels with covariance matrix will be input to a hierarchical clustering algorithm to acquire two cluster labels after topics being merged and split, respectively.

To extract visual words (low level features) of trajectory trees, we construct a codebook for each video scene. We divide the scene image into cells of 10*10 pixels, and quantize the velocity of each trajectory point into 5 bins, as $v \in \{0, 1, 2, 3, 4\}$. Given scene video resolution of $W * H$, the size of the codebook is set to $(W/10) * (H/10) * 5$. With the codebook, we compute a word for each trajectory point with $word = v * (H/10 * W/10) + (x/10) * (H/10) + (y/10)$, where ($x$,$y$) is the coordinate, and $v$ is the velocity bin. After the computation of words for all trajectory points, we represent each tree with a *bag-of-words* [10].

For clustering, the learned topics are merged or split according to the over- or under-segmented relations, which are determined by the covariance $\Sigma$ matrix constructed in the parameter estimation procedure. If an element $\Sigma(i, j)$ of the covariance matrix equals to zero, topics $i$ and $j$ are irrelevant and correspond to a cluster. If $\Sigma(i, j)$ is larger than a threshold, topic $i$ and $j$ are over-segmented, and should be merged into a cluster. If $\Sigma(i, j)$ is smaller than a threshold, topic $i$, $j$ are under-segmented, at least one of the topic $i$, $j$ is shared with multiple clusters. The thresholds to determine over- or under-segmented topics are empirically selected by observing middle level features. The trajectory clustering procedure is detailed in Algorithm 1.

**Algorithm 1** Trajectory Clustering Algorithm

**Input:** Trajectory $i$, the trajectory set $\Im$ /*Without loss generality, we use trajectory $i$ to represent all trajectories.*/
**Output:** The trajectory $i$'s two cluster labels $l_m$, $l_s$

```
Exhaustively seek a neighbor set ε for the trajectory i in set ℑ based in [2]
for each trajectory j in ε do
    γ_j ← spanning tree in [2]
    ψ ← γ_j /*Ψ is the potential tree set */
    EM iterating in BCTM for φ_{γ_j}
    Ω ← φ_{γ_j} /*Ω is the potential topic probability set*/
end for
Initialize z_i=0, l_m=0, l_s=0   /* z_i is the trajectory i's topic label, l_m and l_s are the cluster labels*/
z_i=seek-topic(Ψ,Ω)
seek-cluster(Σ, Ω, z_i, l_m, l_s)   /* Σ is the covariance matrix*/
```

**function seek-topic** (tree set $\Psi$, topic probability set $\Omega$ )

```
γ = argmin_{γ∈Ψ} H(Ω_γ)   /*H(z) = −Σ_z Ω_{γ,z} × log(Ω_{γ,z}) is information entropy, computed over the probabilities of topics for the tree γ*/
z=argmax_z Ω_{γ,z}
return z
end
```

**function seek-cluster** (covariance $\Sigma$, topic probability set $\Omega$, topic label $z_i$, cluster label $l_s$, cluster label $l_m$)

```
initialize Φ_m=∅, Φ_s=∅   /* Φ_m, Φ_s are potential topic sets for merging and splitting, respectively */
for each z_j in topic number k do
    if Σ_{z_i,z_j} > ς_1 then
        Φ_m ← z_j
    end if
    if Σ_{z_i,z_j} < ς_2 then
        Φ_s ← z_j
    end if
end for
l_m = min_z Φ_{m_z}
l_s = argmax_{z∈Φ_s} Ω_{γ,z}
end
```

(a) (b) (c)

Fig. 4. The estimated belief areas in (a) Station [2], (b) Campus [21], and (c) Cross Road [22], respectively.

## IV. Experiments

Experiments are conducted on three datasets, which were collected from the crowded New York's Grand Central station [2], a surveillance camera in a campus [21] and a high-angle camera of a busy cross road [22]. For simplicity, we use the term "Station", "Campus" and "Cross Road" to denote the three datasets. The details of information of datasets are shown in TABLE I. We use the full dataset of Station [2] and Campus [21], but only part of the Cross Road dataset [22], since the scenes in the dataset are not very crowded.

TABLE I. Information of Datasets

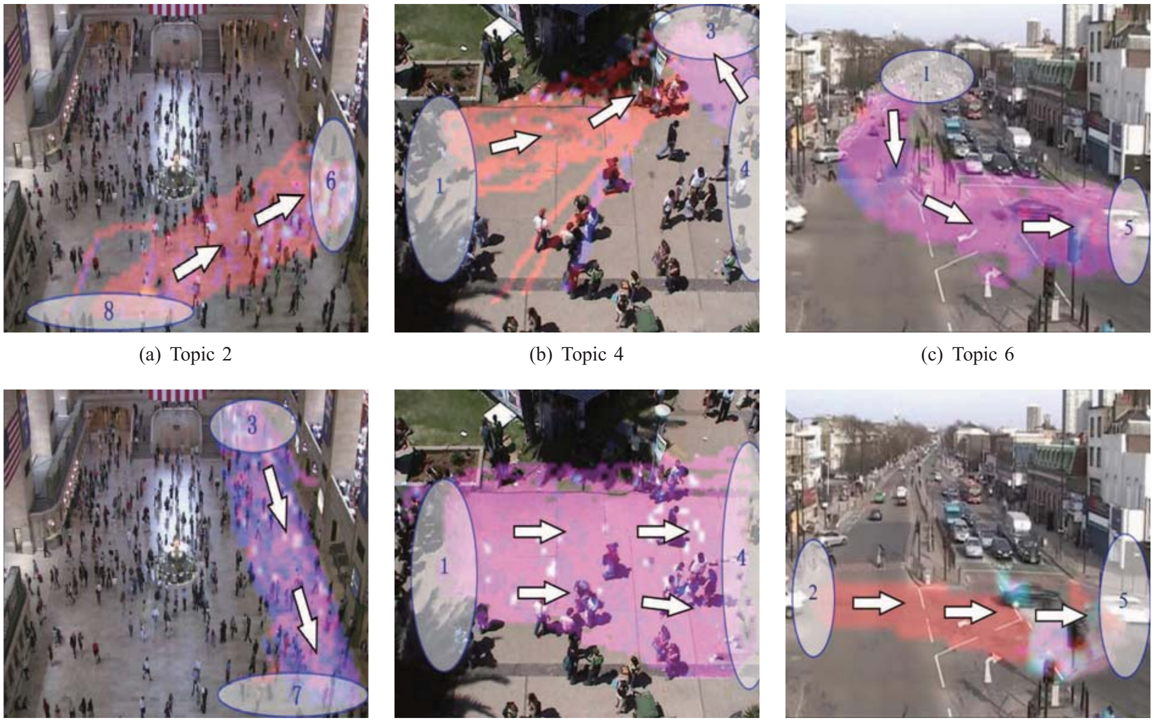
| Dataset | Resolution | Time-Length | Codebook size | Trajectories |
|---|---|---|---|---|
| Station [2] | 720*480 | 1800s | $72 \times 48 \times 5$ | 47866 |
| Campus [21] | 360*288 | 216s | $36 \times 29 \times 5$ | 32455 |
| Cross Road [22] | 720*576 | 373s | $72 \times 58 \times 5$ | 51136 |

TABLE II. Information of Labeled Trajectory Pairs

| Dataset | Completeness | Correctness | Average length |
|---|---|---|---|
| Station [2] | 1507 | 2000 | 133 |
| Campus [21] | 500 | 500 | 159 |
| Cross Road [22] | 1000 | 1000 | 185 |

The parameter $\beta$ contains the discriminative middle level features learned by the BCTM. In Fig. 5, it can be seen that middle level features learned by the BCTM are some trajectory regions, which correspond to the paths in the scenes and are obviously discriminative. For the two clusters ($l_m$ and $l_s$ calculated by Algorithm 1), the trajectories will be assigned to the cluster whose corresponding topic's probability is the highest. Some representative clusters of trajectories are shown in Fig. 6, which are learned by our clustering algorithms with "BCTM" features. Different colors denote different clusters.

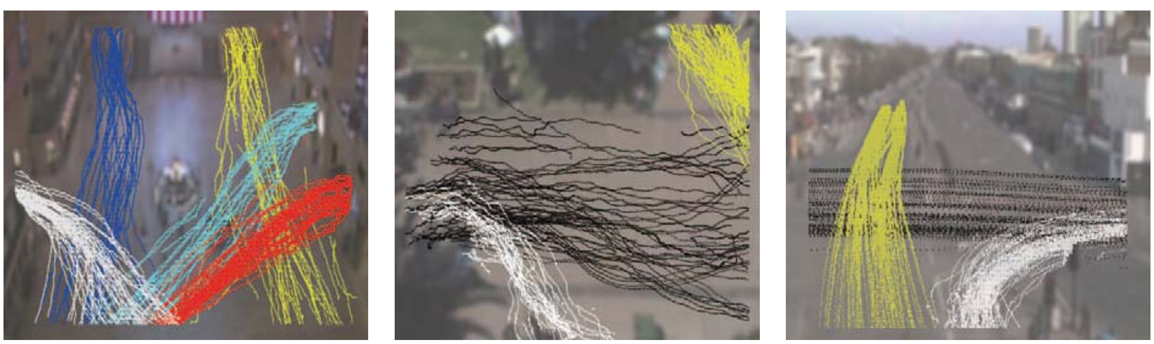
We use correctness and completeness [2] to measure the clustering accuracies. Completeness measures now accurately the trajectories from the same clusters are clustered together. Correctness measures now accurately the trajectories from the different clusters are divided. Therefore, if all trajectories are clustered into one single cluster, the completeness is 100% and the correctness is 0%, and vice versa. We manually label trajectories as ground truth in three different datasets, shown in TABLE II. As discussed above, a trajectory gets two cluster labels after its corresponding topics are hierarchically clustered. In this experiment, completeness accuracy is primarily related to the over-segmented topics and correctness accuracy is primarily related to the under-segmented topics. The merged cluster label $l_m$ is for completeness accuracy, and the split cluster label $l_s$ is for correctness accuracy. The comparisons of different contributions are listed in Fig. 7, 8. We compared our work to the Spectral Clustering (SC) approach [8] in which we implemented. We used a linear interpolation to align the trajectories and measure the similarities with the Euclidean distance.Fig. 7 shows that our approach outperforms other three approaches on all the three datasets, with a significant improvement of completeness accuracies. It can be seen that the more topics we have, the higher completeness accuracy is. With more topics to share the data's clusters, BCTM can learn more discriminative middle level features, so we can choose a more reasonable threshold to identify over-segmented topics. This can be done on the visualized discriminative middle level features, as shown in Fig. 5. As most used trajectories are short and mixed in Station, SC often fails to cluster them in Fig. 7(a). In Fig. 7(b), as most of the trajectories are lying between area 1 and 4 in Fig. 4(b), other three approaches failed to perform trajectory clustering. In contrast, BCTM performs well. In Fig. 7(c), BCTM also performs better than other compared approaches even when the objects are vehicles.As shown in



(a) Topic 2 (b) Topic 4 (c) Topic 6

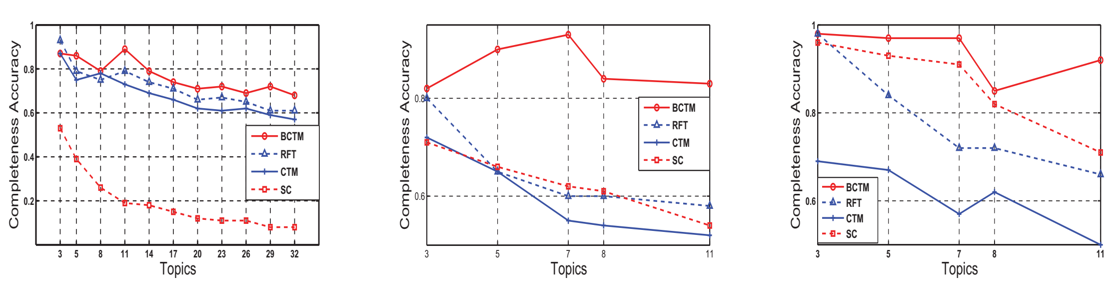
(d) Topic 4 (e) Topic 6 (f) Topic 8

Fig. 5. Representative middle level features learned by BCTM, arrows denote the directions of the paths. The two circles on each path denote the learned belief areas [20]. (Better view in color version)



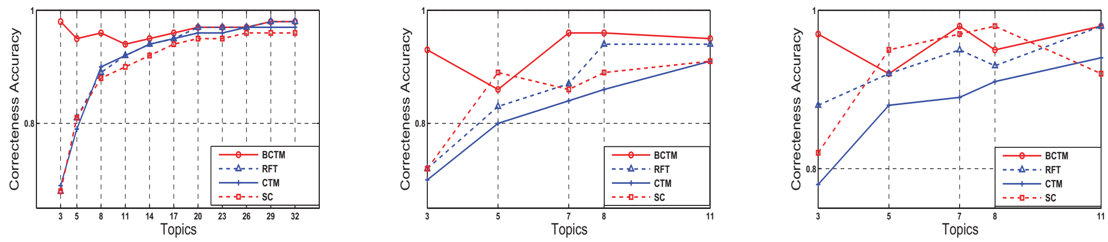
(a) Station dataset (b) Campus dataset (c) Cross Road dataset

Fig. 6. Representative trajectory clusters calculated by the proposed approach. (Better view in color version)




(a) Station dataset (b) Campus dataset (c) Cross Road dataset

Fig. 7. Completeness accuracy.




(a) Station dataset (b) Campus dataset (c) Cross Road dataset

Fig. 8. Correctness accuracy.

Fig. 8, the proposed approach has better correctness accuracies compared to other approaches, except for the SC approach with three and eight topics in Cross Road and five topics in Campus. The reason that the SC approach can perform better is that the scenes are not as crowded as Station, and long and complete trajectories could be obtained with the object tracking algorithm. They were clustered well with the SC approach. However, with the increasement of topic number, the accuracy of the SC approach drops.

## V. Conclusion

We have proposed a belief based Correlated Topic Model (BCTM) to learn trajectory middle level features for trajectory analysis and clustering. By constructing a scene prior based joint Gaussian distribution over the topics, BCTM could effectively reflect the relations between topics and clusters, and learn discriminative middle level trajectory features. The middle level features are effectively implemented into a hierarchical clustering algorithm for trajectory clustering. We validated the effectiveness of the proposed approach and compared it with three recent approaches on three datasets. Experiments and comparisons show that the proposed approach significantly improves the trajectory clustering performance. In addition, the performance is stable under different parameters.

## Acknowledgment

This work is supported in Part by National Basic Research Program of China (973 Program) with Nos. 2011CB706900, 2010CB731800, and National Science Foundation of China with Nos. 61039003, 61271433 and 61202323.

## References

[1] B. T. Morris and M. M. Trivedi, "A survey of vision-based trajectory learning and analysis for surveillance," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 18, no. 8, pp. 1114–1127, 2008.

[2] B. Zhou, X. Wang, and X. Tang, "Random field topic model for semantic region analysis in crowded scenes from tracklets," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 3441–3448, 2011.

[3] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Trans. Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 34, no. 3, pp. 334–352, 2004.

[4] B. Morris and M. Trivedi, "Learning trajectory patterns by clustering: Experimental studies and comparative evaluation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 312–319, 2009.

[5] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *Acm Computing Surveys*, vol. 38, no. 4, p. 13, 2006.

[6] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering." *In Advances in Neural Information Proccessing Systems*, vol. 14, pp. 585–591, 2001.

[7] X. Wang, K. Tieu, and E. Grimson, "Learning semantic scene models by trajectory analysis," *Proc. European Conf. Computer Vision*, pp. 110–123, 2006.

[8] W. Hu, D. Xie, Z. Fu, W. Zeng, and S. Maybank, "Semantic-based surveillance video retrieval," *IEEE Trans. Image Processing*, vol. 16, no. 4, pp. 1168–1181, 2007.

[9] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. Maybank, "A system for learning statistical motion patterns," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1450–1464, 2006.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[11] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, 2006.

[12] X. Wang, K. T. Ma, G.-W. Ng, and W. E. L. Grimson, "Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models," *International Journal of Computer Vision*, vol. 95, no. 3, pp. 287–312, 2011.

[13] X. Wang, X. Ma, and W. E. L. Grimson, "Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 539–555, 2009.

[14] E. Grimson, X. Wang, G.-W. Ng, and K. T. Ma, "Trajectory analysis and semantic region modeling using a nonparametric bayesian model," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.

[15] D. M. Blei and J. D. Lafferty, "A correlated topic model of science," *The Annals of Applied Statistics*, pp. 17–35, 2007.

[16] M. Rodriguez, J. Sivic, I. Laptev, and J.-Y. Audibert, "Data-driven crowd analysis in videos," *Proc. IEEE Conf. Computer Vision*, pp. 1235–1242, 2011.

[17] N. Rasiwasia and N. Vasconcelos, "Latent dirichlet allocation models for image classification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2665–2679, 2013.

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

[19] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," *Machine Learning*, vol. 37, no. 2, pp. 183–233, 1999.

[20] C. Stauffer, "Estimating tracking sources and sinks," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 4, pp. 35–35, 2003.

[21] F. Solera, S. Calderara, and R. Cucchiara, "Structured learning for detection of social groups in crowd," *Proc. IEEE Conf. Advanced Video and Signal Based Surveillance*, pp. 7–12, 2013.

[22] C. C. Loy, T. M. Hospedales, T. Xiang, and S. Gong, "Stream-based joint exploration-exploitation active learning," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1560–1567, 2012.