
Article:
Al Ghamdi, M. and Gotoh, Y. orcid.org/0000-0003-1668-0867 (2020) Graph-based topic models for trajectory clustering in crowd videos. Machine Vision and Applications, 31. 39. ISSN 0932-8092

https://doi.org/10.1007/s00138-020-01092-3

Machine Vision and Applications manuscript No.
(will be inserted by the editor)

Graph-based Topic Models for Trajectory Clustering in Crowd Videos

Manal Al Ghamdi · Yoshihiko Gotoh

Received: date / Accepted: date

Abstract Probabilistic topic modelings, such as latent Dirichlet allocation (LDA) and correlated topic models (CTM), have recently emerged as powerful statistical tools for processing video content. They share an important property, *i.e.*, using a common set of topics to model all data. However such property can be too restrictive for modeling complex visual data such as crowd scenes where multiple fields of heterogeneous data jointly provide rich information about objects and events. This paper proposes graph-based extensions of LDA and CTM, referred to as GLDA and GCTM, to learn and analyze motion patterns by trajectory clustering in a highly cluttered and crowded environment. Unlike previous works that relied on a scene prior, we apply a spatio-temporal graph (STG) to uncover the spatial and temporal coherence between the trajectories of crowd motion during the learning process. The presented models advance the conventional approaches by integrating a manifold-based clustering as initialization and iterative statistical inference as optimization. The output of GLDA and GCTM are mid-level features that represent the motion patterns used later to generate trajectory clusters. Experiments on three different datasets show the effectiveness of the approaches in trajectory clustering and crowd motion modeling.

Keywords clustering · crowd videos · graph · manifold embedding · topic modeling

M. Al Ghamdi
Department of Computer Science, Umm Al-Qura University, Saudi Arabia
E-mail: maalghamdi@uqu.edu.sa

Y. Gotoh
Department of Computer Science, University of Sheffield, United Kingdom
E-mail: y.gotoh@sheffield.ac.uk

1 Introduction

Trajectory clustering and analysis of crowd motion have been vital components of various applications in public surveillance, such as flow estimation. The goal is to analyze individuals' movements by a trajectory associated with a cluster label, thus representing individuals' pathways. A highly crowded scene is particularly challenging because of the density, heavy occlusions and variations in the view. Additionally interaction between individuals can lead to misdetection of body parts [24]. The presence of such challenges makes it difficult to analyze movements using conventional techniques such as background subtraction and motion segmentation, although they may work effectively for less-crowded scenes.

To overcome the shortcomings of conventional techniques, motion patterns have been investigated for processing crowd scenes. In such scenarios objects are represented by a small number of pixels; there is thus ambiguity in appearance caused by the dense packing [18]. Defining the motion patterns in a crowd scene becomes a key to the problem. Examples of motion pattern techniques include scene structure-based force models [3] and the Bayesian framework with spatio-temporal motion models [12]. Typically these models are based on the assumption that the objects move coherently in one direction throughout a video. This is a major shortcoming, as they fail to represent complex crowd scenes with multiple dominant crowd behaviors in multiple locations.

1.1 Related Work

Trajectory clustering is fundamental to solve the multi-object tracking problem in various applications such as crowd analysis and video surveillance. In many applications

a vast amount of trajectories and motion patterns are extracted and clustered into groups without manually labeling the data based on various methods including distance-based clustering [28], waypoint clustering [11], tree-based clustering [36], grid-based clustering [19] and kernel clustering [35]. Despite the vast literature [26, 30, 14], this problem still remains a challenging especially in very crowded scenes with occlusions leading to false detection.

Based on the social force model [23], Pellegrini *et al.*[21] proposed a linear trajectory avoidance (LTA) model to predict the optimal path for individuals that prevents collisions with each other and the obstacles. They performed experiments using non-crowded scenes with lower applicability for collision than to dense crowded scenes. Lin *et al.* [16] detected motion trajectories in crowd scenes by processing the flow fields. They applied a two-step clustering process to define semantic regions which were used later to recognize pre-defined activities in the crowd. Lu *et al.* [17] extracted motion trajectories to investigate characteristics of pedestrians in an unstructured scene. In their work trajectories were firstly represented as a four-dimensional vector, then clustered using the fuzzy c-means (FCM) algorithm to form motion patterns. Sharma and Guho [28] proposed a two-step trajectory clustering approach (TCA) to segmenting crowd flow patterns; a trajectory extraction step to detect and track blocks or regions in the video, followed by a clustering step that utilized the shape, the location and the density of the trajectory in the neighborhood. Rabiee *et al.* [22] detected abnormal behaviors from crowd scenes using a spatio-temporal tracklet based descriptor extracted from 3D patches. The tracklets were extracted by tracking randomly selected points in video frames within a short period of time. Using the orientation and magnitude of extracted tracklets, one-dimensional descriptors were derived and fed into one-class support vector machine (SVM) classifier for abnormality detection. Recently, Burceanu and Leordeanu [9] proposed a neural network object tracker with two pathways; the FilterParts and the ConvNetPart. The first pathway is robust to background noises while the second one is robust to object appearance changes over time. The object's next moved tracking is determined based on the vote for center maps from the two pathways.

Many works have been proposed for trajectory clustering based on mid-level features learning. These features are usually observed as pathways defined by individuals' movements, thus designed to map the segments of trajectories from a low-level feature space to their clusters [39]. A trajectory for mid-level features can be learnt using hierarchical latent variable Bayesian models, such as latent Dirichlet allocation (LDA) [6] and correlated topic models (CTM) [5]. These models are known as 'topic models', adopted from the text-processing field. They often have hierarchical structures where latent variables lie at multiple levels. Using these models documents are represented by trajectories and visual words are defined by observations of object trajectories. With these approaches the learnt topics represent mid-level features of trajectories.

CTM was adopted to the video-processing domain by Rodriguez *et al.* [24] as a mid-level feature to represent multiple motion behaviors in one scene. Their tracker was weighted to predict a rough displacement using a codebook generated from all the moving pixels in a scene, along with the learnt high-level behavior. Although CTM was an effective model, it only processed motions at each spatial location and disregarded the temporal correlation between sequential motions that could naturally occur in crowd scenes, hence it could not create discriminative mid-level features for multiple clusters. Rodriguez *et al.*[25] proposed a data-driven crowd analysis algorithm that learn the crowd behaviour priors from large database using the CTM. The crowd patches in the testing videos were then matched to the database using local and global Scene Matching. Their method based on the assumption that all crowd behaviours were learnt from the database. Thus, it may fail if a tested video involves any behaviours that have no corresponding matches in the database. A scene prior belief based correlated topic model (BCTM) [39] was then proposed to construct a mid-level features for trajectory clustering. A feature tracker was firstly employed to generate trajectories. A spanning tree method was then used to define the initial clusters. The mid-level features were generated using BCTM followed by a hierarchical clustering algorithm to produce the final clusters. Their experiment showed that BCTM as a trajectory clustering method outperformed CTM, but it could only be applied if a scene prior was available.

Zhou *et al.* [38] proposed a random field topic (RFT) model to perform trajectory clustering in a crowd scene. It extended the LDA models by integrating a scene prior and using a Markov random field (MRF). RFT significantly improved the clustering performance over LDA models; however the performance could drop in crowd scenes with correlated topics where topics were shared with multiple clusters and where clusters were also shared with multiple topics. Chen *et al.* [10] presented a patch-based topic model for group detection. They used the feature points distribution over the orientation space as a patch-level descriptor, which was then fed into the LDA model to learn the semantic motion within each patch. Their model utilized MRF as a prior to enforce the spatial coherence and to cluster the features based on a prior of the corresponding patch.

1.2 This Work

Although recent approaches offered effective solutions, most of them ignored the temporal relationship within crowd scenes and the distribution of data. Instead they required

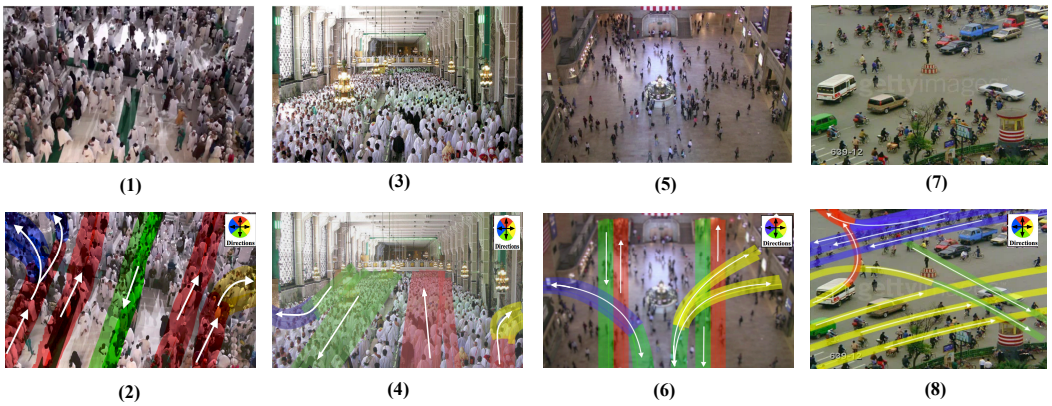


Fig. 1 Sample frames (the first row) and the motion patterns (the second row) of crowded indoor scenes, from Al-Masjid Al-Haram (S1), (S2) in Mecca [4], Grand Central Station in New York [38] and the Collective Motion (CUHK) Database [37]. (Seen better in color.)

complex parameter estimation and variable inference procedures. This paper presents two graph-based topic models, graph-based latent Dirichlet allocation (GLDA) and the graph-based correlated topic model (GCTM) [2], for analyzing crowd motion and clustering trajectory in a complex crowd scene. Both models extended the conventional models by integrating a spatio-temporal graph (STG) to enforce the spatial and temporal coherence between trajectories during the learning process. The goal of this work is to address the problem of trajectory clustering and motion pattern (or movement direction) analysis in high-density crowds without using any prior knowledge of the motion pattern of a scene. Different from previous works, both GLDA and GCTM have a manifold-based cluster initialization step, that is followed by iterative optimization with Bayesian inference. The initialization step helps our models to generate topics which means motion patterns (mid-level features), that effectively reflect data distribution and cluster information. After the iterative optimization the generated topics are discriminative where different trajectories are clustered separately in the manifold space.

This paper is an extended version of our earlier work published in [2], which presented the GCTM to learn and analyse motion patterns by trajectory clustering in a highly cluttered and crowded environment. In this paper, we present the GLDA as a spatio-temporal graph-based extension of the LDA, which is widely-used model in the family of statistical topic models and is more suitable over the CTM specially when documents are long and the correlation between topics is not important [15]. Additionally, we present a comprehensive analysis of the results obtained using both GCTM and GLDA on three crowd datasets with a range of diversities to show the effectiveness of the extended models. Our other work in [2] extended the CMT using the spatio-temporal graph followed by the k-nearest neighbourhood (kNN) clustering method without dimensionality reduction. In that paper [1], tracklets cluster prediction was performed based on the minimum entropy. While in the method presented in [3] and in this paper, the GCTM does trajectory clustering after dimensionality reduction and the tracklets cluster prediction are performed using the maximum likelihood. Both previous papers [1,2] presented comparable results and outperformed the related approaches. The GCTM presented in [2] run faster thanks to the manifold embedding.

The presented methods started by apply the Kanade-Lucas-Tomasi (KLT) tracker [31] to extract trajectories points, that are used later by the locality-constrained linear coding (LLC) technique [34] to generate a set of visual codes as low-level features. The STG is then constructed to uncover the spatio-temporal relations between the trajectories and projected to lower-dimensional space to initialize clusters in a manifold embedding space. Using cluster labels, topics are learnt by GLDA and GCTM for final trajectory clustering. Experiments are performed on three different video datasets; one collected from multiple indoor locations at crowded Al-Masjid Al-Haram [4], the second one collected at the Grand Central Station in New York [38] and the third one collected from different indoor and outdoor crowd scenarios (Figure 1).

The remainder of the paper is organized as follows: the proposed graph-based models are introduced in Section 2. The initial and final trajectory clustering techniques are presented in Section 3. Datasets and experimental setup are presented in Section 4, which are followed by results and discussion. Finally Section 6 concludes the paper.

2 Graph-based Topic Models

LDA assumes that a word in topics contains a multinomial distribution, that a document contains multiple topics and that the ration of topics varies following a Dirichlet distribution. CTM follows the same generative process of LDA

but, instead of the Dirichlet distribution, it uses the logistic normal distribution to capture the correlation among topics. This section presents the approach to learning mid-level features (topics) as motion patterns (movement direction) using GLDA (Section 2.2) and GCTM [2] (Section 2.3). We show that, by extending LDA and CTM to utilize initial clusters based on spatio-temporal graph, we are able to greatly simplify the training algorithm, thus creating distinctive topics for clustering without using any scene prior. This means that different trajectories will have different clusters in the manifold space.

2.1 Notation

Figures 2(a) and (d) show graphical representations of the conventional LDA [6] and CTM [5] that were originally developed in the text-processing field. Both models assume that M, N and K denote the number of documents, the number of words in a document and the number of hidden variables (or 'topics') in the model, respectively. The circles in the figures are random variables or model parameters, and the edges specify the probabilistic dependencies (or the conditional independences) among them; boxes, with M, N and K, are compact notations for multiple instances of the variables or parameters. Shaded variables represent the observed variables, while unshaded variables indicate the latent variables.

Corpus, document, topic and words (for text data) in the conventional models are replaced with pathway, trajectory, motion pattern (or movement direction) and visual codes (for video data) in the graph-based models. The topic mixture of a document corresponds to a set of different motion patterns in a trajectory. The graph-based models learn crowd motion by clustering trajectories. The graphical representations of GLDA and GCTM are presented in Figures 2(b) and (e). Observed visual codes (low-level features) and initial clusters are the inputs for both models. Section 3 describes the construction of the visual codes and initial clusters as low-level features.

We begin with some notations and definitions for parameters used with both models:

- M is the number of trajectories in the pathway, each of which is modeled as a mixture of K topics. $m = 1, \ldots, M$ is the index of an individual trajectory in the pathway.
- N is the total number of visual occurrences in a trajectory m. $n = 1, \ldots, N$ is the index of a visual code occurrence in a document m.
- K is the number of hidden topics in the model, where each topic is a distribution over a code set given by a hyper-parameter β_k.
- $c \sim p(c \mid \eta)$ where $c = 1, \ldots, C$ is an initial cluster defined for each trajectory. C is the total number of initial clusters and η is a C-dimensional vector of a multinomial distribution.
- π_m (or π) in GLDA is a discrete variable sampled from a Dirichlet distribution for choosing the topic $p(\pi_m \mid \alpha, c)$.
- θ_m (or θ) in GCTM is a continuous variable sampled from a Gaussian distribution for choosing the topic $p(\theta_m \mid \mu, \Sigma, c)$.
- μ is a K-dimensional vector and Σ is a $K \times K$ covariance matrix, parameters of a multivariate Gaussian process.
- α is a $C \times K$ matrix, and α_c is a K-dimensional Dirichlet parameter conditioned on the topic c.
- $z_{m,n}$ (or z_n) is a hidden variable assigned to a visual code x_n drawn from a multinomial distribution.
- $x_{m,n}$ (or x_n) is a visual code n in the trajectory m.

2.2 GLDA: Graph-based Latent Dirichlet Allocation

LDA assumes that there is a different discrete distribution π for each document to generate topics for words and that all documents share a Dirichlet prior α. In Figure 2(a), π_m (or π) is a K-dimensional vector representing a topic prior for each document; $z_{m,n}$ (or z_n) is a hidden variable, following a parameterized multinomial distribution $Mult(\pi)$; $x_{m,n}$ (or x_n) is the random variable whose value is the observed word (*i.e.*, 'feature'); and β is a hyper-parameter corresponding to the mid-level features. The generative process of LDA is outlined as follows:

- Choose $\pi \sim Dirichlet(\alpha)$.
- For each visual word x_n for $n \in \{1, \ldots, N\}$:
  1. Choose a topic $z_n \mid \pi$ according to $Mult(\pi)$;
  2. Choose a word $x_n \mid \{z_n, \beta_{1:K}\}$ according to $x_n \sim p(x_n \mid z_n, \beta)$.

Using this model the document probability, given a topic variable π, a word x and an individual topic assignment z, is expressed as

$$P(x, z, \pi \mid \beta, \alpha) = p(\pi \mid \alpha) \prod_{n=1}^{N} P(z_n \mid \pi) P(x_n \mid z_n, \beta) \tag{1}$$

Note that the topic-level information given by π and z is hidden, while the word-level representation is observed.

GLDA requires both observed visual words and initial clusters as inputs to the model. Given the parameters α, η and β, the joint probability of GLDA, with a set of N topics

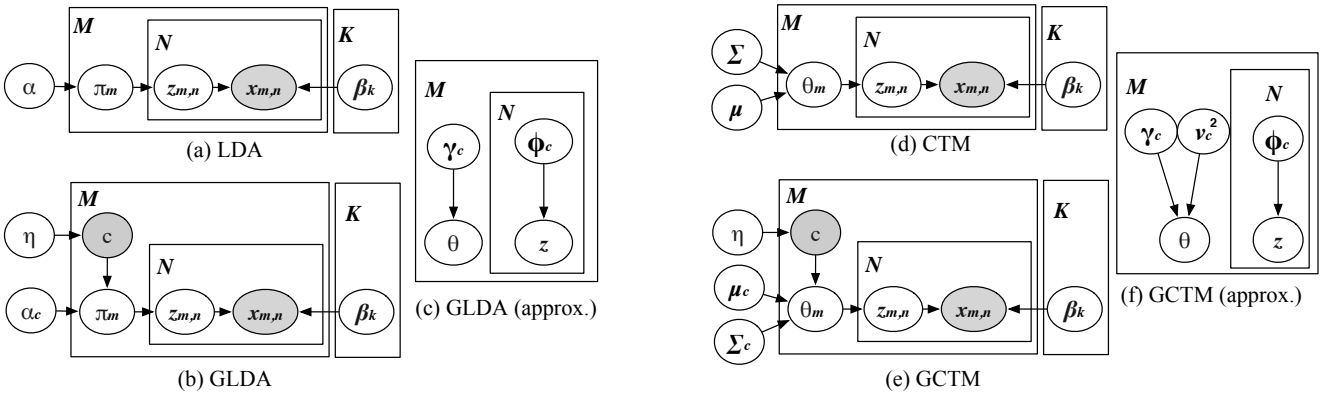


Fig. 2 Graphical representation of (a) LDA, (b) GLDA, (c) its approximate distribution, (d) CTM, (e) GCTM and (f) its approximate distribution.

z, a set of N visual codes x and a cluster c, is

$$p(x, z, \pi, c \mid \eta, \beta, \alpha) = p(c \mid \eta)p(\pi \mid \alpha, c) \prod_{n=1}^{N} p(z_n \mid \pi)p(x_n \mid z_n, \beta) \quad (2)$$

where

$$p(x_n \mid z_n, \beta) = \prod_{k=1}^{K} p(x_n \mid \beta_k) \quad (3)$$

$$p(\pi \mid \alpha, c) = \prod_{c=1}^{C} Dirichlet(\pi \mid \alpha_c) \quad (4)$$

$$p(c \mid \eta) = Mult(c \mid \eta) \quad (5)$$

$$p(z_n \mid \pi) = Mult(z_n \mid \pi) \quad (6)$$

The distribution of $p(c \mid \eta)$ is always assumed as a fixed uniform distribution $p(c) = 1/C$. Therefore we leave out the estimation of η. The log probability for x is given as

$$p(x \mid \alpha, \beta, c) = \int p(\pi \mid \alpha, c) \left(\sum_{z} \prod_{n=1}^{N} p(x_n \mid z_n, \beta)p(z_n \mid \pi) \right) d\pi \quad (7)$$

We use the variational breaking algorithm in [6] to estimate parameters of the GLDA. Figure 2(c) is the graphical representation for the approximate distribution for GLDA. We now have

$$\log p(x \mid \alpha, \beta, c) = L(\gamma_c, \phi_c; \alpha_c, \beta) + KL\{q(\pi, z \mid \gamma_c, \phi_c) \parallel p(\pi, z \mid x, \alpha_c, \beta)\} \quad (8)$$

where $KL\{\cdot\}$ implies the Kullback-Leibler distance. We iteratively maximize the term $L(\cdot)$, instead of $p(x \mid \alpha, \beta, c)$, which results in the minimum of the difference between distributions in Figure 2(b) and Figure 2(c). Further details of computation is found in [6]. We give modified parameters and variables as

$$\phi_{ki}^{c} \propto \exp(\gamma_k^c)\beta_k \quad (9)$$

$$\beta_k \propto \sum_i \phi_{k,i}^{c} n_i \quad (10)$$

where m is used to index the trajectory, i to index the word and k to index a topic. $\phi_{k,i}$ denotes the probability that the ith word belongs to the kth topic, n_i is the word count and β_k is the kth topic's representation in the word space.

2.3 GCTM: Graph-based Correlated Topic Model

In the Dirichlet distribution the components are considered independent, thus each topic cannot have a relation with other topics. This independence practically prevents occurrence of a word in other topics — that is, if topics are fully independent, a word in one topic cannot appear in other topics. In order to address the issue CTM assumes that each document is a mixture of words given a set of hidden topics, and in turn each topic is determined by a distribution over the entire vocabulary. It employs more flexible logistic normal distribution to represent a covariance structure among the components. The formulation of GCTM is analogous to the one for GLDA. It is presented below to contract the similarity and the difference between GLDA and GCTM.

In Figure 2(d), θ_m (or θ) is a K-dimensional vector, specifying a topic prior for each document; $z_{m,n}$ (or z_n) is a hidden variable, following a parameterized multinomial distribution $Mult(\theta)$; $x_{m,n}$ (or x_n) is a random variable whose value is an observed word (*i.e.*, 'feature'); and β is a hyperparameter corresponding to the mid-level features. Finally μ and Σ are the mean and the covariance matrix of the multivariate normal distribution. The generative process of CTM is outlined as follows:

- Draw $\theta \mid \{\mu, \Sigma\} \sim \mathcal{N}(\mu, \Sigma)$.

– Draw a document-specific topic component π as $\pi = \frac{\exp(\theta)}{\sum_{i=1}^{K}\theta_i}$.
– For each visual word x_n for $n \in \{1,\ldots,N\}$:
  1. Assign a topic $z_n \mid \theta$ according to $Mult(\pi)$;
  2. Choose a word $x_n \mid \{z_n, \beta_{1:K}\}$ according to $p(x_n \mid z_n, \beta)$.

Using this model the document probability, given a topic variable θ, a word x and an individual topic assignment z, is expressed as

$$p(\theta, z, x \mid \mu, \Sigma, \beta) = p(\theta \mid \mu, \Sigma) \prod_{n=1}^{N} p(z_n \mid \theta) p(x_n \mid z_n, \beta) \tag{11}$$

Note that the topic-level information given by θ and z is hidden, while the word-level representation is observed.

GCTM requires both observed visual words and initial clusters as inputs to the model. Given the parameters Σ, μ, η and β we can now write a full set of generative equations for the GCTM model. The joint probability of a topic mixture θ, a set of N topics z, a set of N visual codes x and the cluster c is

$$P(x, z, \theta, c \mid \eta, \beta, \mu, \Sigma) = p(c \mid \eta) p(\theta \mid \mu, \Sigma, c) \prod_{n=1}^{N} p(z_n \mid \theta) p(x_n \mid z_n, \beta) \tag{12}$$

where

$$p(\theta \mid \mu, \Sigma, c) = \prod_{c=1}^{C} \mathcal{N}(\theta \mid \mu_c, \Sigma_c) \tag{13}$$

$$p(c \mid \eta) = Mult(c \mid \eta) \tag{14}$$

$$p(z_n \mid \theta) = Mult(z_n \mid \theta) \tag{15}$$

The log probability for x is given as

$$p(x \mid \mu, \Sigma, \beta, c) = \int p(\theta \mid \mu, \Sigma, c) \left(\sum_{z} \prod_{n=1}^{N} p(x_n \mid z_n, \beta) p(z_n \mid \theta) \right) d\theta \tag{16}$$

In order to estimate parameters for GCTM, we used parts of video sequences as training data and adopt the variational expectation maximization (EM) algorithm to do variable inference and parameter estimation [5]. Figure 2(c) is the graphical representation of the approximate distribution for GCTM where $\gamma_{M\times K}$, $v_{M\times K}$ and Φ are variational parameters. The log-likelihood for a document m is given by

$$\log p(x \mid \mu, \Sigma, \beta, c) = L(\gamma_c, v_c, \phi_c; \mu_c, \Sigma_c, \beta) + KL\{q(\theta, z \mid \gamma_c, v_c, \phi_c) \parallel p(\theta, z \mid x, \mu_c, \Sigma_c, \beta)\} \tag{17}$$

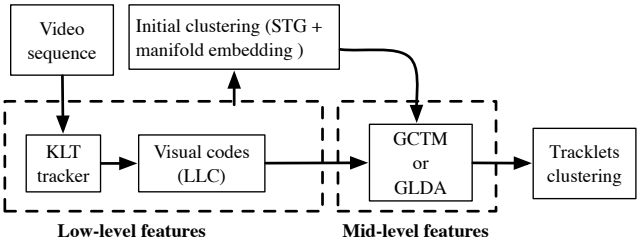


Fig. 3 The framework for crowd behavior modeling using GLDA or GCTM.

As before we iteratively maximize the term $L(\cdot)$ which results in the minimum of the difference between the distribution in Figure 2(e) and Figure 2(f). Modified parameters and variables are given as

$$\phi_{ki}^{c} \propto \exp(\gamma_k^c)\beta_k \tag{18}$$

$$\beta_k \propto \sum_i \phi_{ki}^{c} n_i \tag{19}$$

$$\mu = \frac{1}{M}\sum_m \gamma_m^c \tag{20}$$

$$\Sigma = \frac{1}{M}\sum_m \left\{ diag(v_m^c) + (\gamma_m^c - \mu_c)(\gamma_m^c - \mu_c)^\top \right\} \tag{21}$$

where m is used to index the trajectory, i to index the word and k to index a topic. ϕ_{ki} denotes the probability that the ith word belongs to the kth topic, n_i is the word count and β_k is the kth topic's representation in the word space.

3 Trajectory Clustering

The first step for trajectory clustering is to generate low-level features by extracting trajectory segments and representing them with a collection of visual codes (*i.e.*, words). Secondly, a spatio-temporal graph is applied on the visual codes to uncover spatio-temporal relations among trajectories and embed them in the lower dimensional space to identify initial clusters. Given initial clusters and a set of visual codes, mid-level features are learnt by GLDA or GCTM (Sections 2.2 and 2.3) to produce the final trajectory clustering. The framework is shown by a flow chart in Figure 3.

3.1 Low-level Features

Given a video sequence, the KLT tracker [31] is applied to calculate M trajectories. The LLC algorithm is employed to represent each trajectory with a set of visual codes X as low-level features. LLC is a coding scheme proposed by Wang *et al.* [34] to project features onto their respective local coordinate systems and encode them using fewer codebook basis in the high-dimensional feature space.

Given a trajectory m with a set of points $m = \{t_1, \ldots, t_N\}$, a set of codes $X = \{x_1, \ldots, x_N\}$ are derived by firstly constructing a neighborhood graph based on the geodesic distances between the trajectory points and the codebook, then computing the shortest path performing a kNN search, and finally solving the following constrained least square fitting problem:

$$\min_X \sum_{i=1}^{N} \|t_i - Bx_i\|^2 + \lambda \|d_i \odot x_i\|^2 \quad st.\ 1^\top x_i = 1\ \forall i \tag{22}$$

where $\odot$ implies the element-wise multiplication, B is a codebook, and λ is a sparsity regularization term. Furthermore, '$1^\top x_i = 1\ \forall i$' means the shift-invariant requirements for the LLC code. The locality-constrained parameter d_i represents each basis vector with different freedom based on its shortest path to the trajectory point t_i. The final step uses the multi-scale max pooling [27], where a set of codes computed for each trajectory are grouped together to create the corresponding pooled representation X.

3.2 Initial Clustering

To obtain the initial clusters C for the trajectories, the STG algorithm [2] is applied to uncover spatio-temporal relations among trajectories. The structure in the high-dimensional space is transferred to a spatio-temporal distance graph of nodes with LLC representations. The method reconstructs the order of the LLC representations based on their spatio-temporal relationship and recalculates distances along them to ensure the shortest distance. Firstly the similarity matrix R is calculated between the LLC representations using the Euclidean distance. The value of R_{ij} defines the distance between X_i and X_j of two trajectories ($i, j = 1, \ldots, M$). Then for each instance X_i ($i = 1, \ldots, M$):

1. L codes, closest to X_i, are connected. They are referred to as spatial neighbors S_{X_i}:

$$S_{X_i} = \left\{ X_{j1}, \ldots, X_{jL} \mid \underset{j}{\operatorname{argmin}}^{L}(R_{ij}) \right\} \tag{23}$$

   where $\underset{j}{\operatorname{argmin}}^{L}$ implies L node indices with the shortest distances to X_i.
2. Another L chronologically ordered neighbors around X_i are set as temporal neighbors T_{X_i}:

$$T_{X_i} = \left\{ X_{j-\frac{L}{2}}, \ldots, X_{j-1}, X_{j+1}, \ldots, X_{j+\frac{L}{2}} \right\} \tag{24}$$

3. Optimally T_S is selected from temporal neighbors of spatial neighbors as:

$$T_{S_{X_i}} = \left\{ T_{X_{j1}} \cup \ldots \cup T_{X_{jL}} \right\} \cap T_{X_i} \tag{25}$$

4. The union between spatial and temporal sets represents spatio-temporal neighbors U_{X_i} for code X_i:

$$U_{X_i} = S_{X_i} \cup T_{S_{X_i}} \tag{26}$$

The above formulation of U_{X_i} effectively selects X_i's temporal neighbors that are similar, with a good chance, to its spatial neighbors.

Given the spatio-temporal neighborhood graph, a new correlation δ based on the geodesic distances is defined by applying Dijkstra's distance algorithm between the neighboring nodes [32]. The value of δ represents the shortest path distance (neighbor weights) between two nodes X_i and X_j. If node X_j is a spatio-temporal neighbor of X_i and $j \in U_{X_i}$, then $\delta(X_i, X_j) = \omega_{ij}$ and their trajectory has a neighbor relation, otherwise, $\delta(X_i, X_j) = 0$.

The manifold embedding is then modeled by applying the multidimensional scaling (MDS) [7]. It is formed as a transformation of the high-dimensional data in terms of the correlation δ into a new d-dimensional embedded space that best preserves the neighboring relations of the clusters. In the lower dimensional manifold embedding space, a k-means algorithm is adopted to perform clustering and obtain initial trajectory cluster labels.

3.3 Final Clustering

After the mid-level features are learnt and the topic probabilities of trajectories are computed, each trajectory has a set of K topics to choose from. A topic label with the highest probability is assigned to the trajectory. Given a new trajectory m with an unknown path, LLC representation X is firstly defined with N visual codes and the probability of each cluster is computed with GLDA as:

$$p(c \mid x, \alpha, \beta, \eta) \propto p(x \mid c, \alpha, \beta) p(c \mid \eta) \propto p(x \mid c, \alpha, \beta) \tag{27}$$

where α, β and η are parameters learnt by the GLDA model. The decision of the topic is made by comparing the likelihood of X given each cluster label as $\operatorname{argmax}_c p(x \mid \beta, \alpha, c)$ where the term $p(x \mid \beta, \alpha, c)$ is defined as in Eq.(7).

Similarly, with GCTM:

$$\begin{aligned} p(c \mid x, \mu, \Sigma, \beta, \eta) &\propto p(x \mid c, \mu, \Sigma, \beta) p(c \mid \eta) \\ &\propto p(x \mid c, \mu, \Sigma, \beta) \end{aligned} \tag{28}$$

where μ, Σ, β and η are parameters learnt by the GCTM model. The decision of the topic is made by comparing the likelihood of X given each cluster label as $\operatorname{argmax}_c p(x \mid \beta, \mu, \Sigma, c)$ where the term $p(x \mid \beta, \mu, \Sigma, c)$ is defined as in Eq.(16).

dataset	resolution	duration	codebook size	trajectories
Al-Masjid (S1)	960×540	$5,600$ sec	$96 \times 54 \times 4$	87,321
Al-Masjid (S2)	960×540	$3,400$ sec	$96 \times 54 \times 4$	61,760
Station	720×480	$1,800$ sec	$72 \times 48 \times 4$	47,866
CUHK	920×520	$10,300$ sec	$92 \times 52 \times 4$	218,787

Table 1 The resolution, duration, codebook size and the number of extracted trajectories for Al-Masjid (S1), (S2) [4], Grand Central Station [38] and CM [37] datasets.

4 Experiments

We evaluated the graph-based topic models, GLDA and GCTM, using a trajectory clustering task with crowd videos. Once both models were learnt, trajectories were clustered based on the motion pattern (or the movement direction). For each trajectory the topic was assigned to the cluster with the highest likelihood. Three datasets were employed for evaluation:

- Al-Masjid Al-Haram [4] — collected from indoor scenes at the holy mosque of Mecca, Saudi Arabia. This dataset involved a number of difficult problems, such as lighting changes, occlusions, a variety of objects, changes of views and environmental effects. There were two scenes with Al-Masjid videos. The first (S1) was at one of the Tawaf area stairs used to enter or leave the Tawaf. It was a very busy area and needed monitoring to ensure individuals' safety. Multiple pathways could be identified with this scene, including a direct pathway to approach the Tawaf and the left and the right side pathways leading to the seating areas. The second scene (S2) was recorded at the second and the third floors of SAFA and MARWA area, which was a long walkway with two different directions. Along these walkways there were multiple doors used to enter and exit the areas.
- Grand Central Station [38] — collected from the inside of the Grand Central Railway Station in New York, USA. It contained multiple entrances and exits where individuals had multiple pathways to follow. The crowd presented multiple behaviors (or pathways) in various moving directions.
- Collective Motion Database (CUHK) [37] — collected from 62 indoor and outdoor crowded scenes with various densities and scales including streets, shopping malls, airports and parks. It has 413 video clips containing both human and vehicles movements. Manual annotations for the video clips are included in the dataset containing groups or clusters that can be used to evaluate methods for group detection and crowd classification.

For simplicity we denote the datasets as 'Al-Masjid (S1)', 'Al-Masjid (S2)', 'Station' and 'CUHK'. Some details of all datasets are presented in Table 1.

4.1 Experimental Setup

For the low-level feature step, the initial codebook B used for the LLC codes was learnt from a half of the trajectories randomly selected. The $W \times H$ scene was divided into 10×10 cells and the velocities of key-points were quantized into four directions. The pooled representations from the LLC codes were computed for each sub-region (of 4×4, 2×2 and 1×1) and pooled together using the multi-scale max pooling. The number of neighbors was set as $k = 5$ and $\lambda = 500$ in Eq.(22). For the initial clustering we used Elkan's k-means clustering algorithm from the *VLFeat toolbox* [33], which was faster than the standard Lloyd's k-means. The pooled features were concatenated and normalized using the ℓ^2-norm. For STG the similarity matrix was computed using the geodesic distance and the kNN graph was constructed with $L = 20$.

4.2 Evaluation Criteria

For quantitative evaluation of the clustering performance, we adopted correctness and completeness introduced by [20]. We based the evaluation on the criteria that individuals in the same group have a common pathway and form a motion pattern. Thus, the correctness is defined as the accuracy with which a pair of trajectories from different pathways (with the ground-truth) are clustered into different groups. While the completeness is defined as the accuracy with which a pair of trajectories from the same pathway are clustered into the same group. In an extreme case a 100% completeness and a 0% correctness may be achieved when all the trajectories are clustered into a single group. Another extreme is a 0% completeness and a 100% correctness achieved when each trajectory is clustered into a different group. A good clustering algorithm should achieve high scores in both correctness and completeness. We manually labelled 2,500 trajectories for correctness and 1,700 trajectories for completeness with Al-Masjid (S1), 2,000 for correctness and 1,500 for completeness with Al-Masjid (S2), and 2,000 for correctness and 1,500 for completeness with Station. For the CUHK dataset, we used the provided ground-truth and defined 3,500 trajectories for correctness and 2,500 trajectories for completeness.

5 Results and Discussion

Various comparisons have been conducted to evaluate the presented models. Section 5.1 compares the presented models with the related methods reviewed in Section 1.1. The second comparison in Section 5.2 aims to demonstrate the effectiveness of the low-level features, including the KLT tracker and the LLC method used in both GCTM and

GLDA. Section 5.3 validates the effectiveness of the initial clustering, including the STG and the dimensionality reduction used in both GCTM and GLDA, by comparing its performance with other methods.

5.1 The Performances of the Models

Figure 4 presents trajectory clusters for Al-Masjid (S1) video by various approaches, including LDA, CTM[1] [6], RFT[2] (random field topic) [38], GLDA and GCTM. Different colors in the figure represent different clusters (pathways). It can be observed that the graph-based topic models, GLDA and GCTM, were able to produce the cleanest trajectory paths. The other three approaches, LDA, CTM and RFT, failed to perform trajectory clustering well because of their heavy occlusion, which was particularly evident with the side pathways towards the exits. RFT achieved better results for the central pathways in comparison to LDA and CTM. The latter two did not perform well because both of them ignored the temporal correlations. Although they were able to cluster the trajectory segments at one end of the crowd motion (either the starting or the ending position) as one pathway, the other end was not clustered with the same pathway.

Completeness and correctness for LDA, CTM, RFT, GLDA and GCTM are reported in Figures 5 and 6. The results show that GLDA and GCTM outperformed the other three approaches in all three datasets with clear margins. The margins were even wider for completeness when the number of topics was larger. GLDA and GCTM with the STG were able to learn discriminative mid-level features better, even with a large number of topics to share the clusters. The other three approaches did not cluster trajectories well because most of these trajectory segments were short and mixed, thus they were difficult to be clustered. RFT had advanced LDA [6] by accommodating belief priors based on the position and the spatial correlation of trajectories along the video sequence. However the spatio-temporal correlation between trajectories was disregarded. LDA and CTM considered four motion directions at each spatial location, but they ignored the temporal relation between sequential local motions in crowd scenes. CTM performed better than LDA because it considered the correlation between topics. All three methods processed low-level features of the trajectories in the high-dimensional feature space, which was very sparse, making it difficult to directly perform clustering.

Because the Al-Masjid two scenes, S1 and S2, contains more crowded videos than the Station and the CUHK datasets, most of the trajectories generated in the Al-Masjid dataset were short and mixed. It clearly affected adversely the completeness and the correctness accuracies, particularly for LDA, CTM and RFT. In the Al-Masjid (S2) videos, some trajectories were absorbed towards the both sides (blue and yellow trajectories in Figure 1(4)), for which LDA, CTM and RFT failed to perform trajectory clustering. In contrast GLDA and GCTM, with no scene priors, performed well (Figure 5(b) and Figure 6(b)).

Unlike the other two, the CUHK dataset is more challenging because it contains longer clips with various moving objects such as cars and bicycles. It has a number of indoor and outdoor crowd scenarios with different densities. Regardless of types of the object being moved, both GLDA and GCTM were able to identify most of motion patterns and achieved the highest performances in Figure 5(d) and Figure 6(d). Consideration of the temporally coherent motions helped the graph-based models to define the movement directions in various crowd density in the scenes, while the other methods detected the motions frame by frame separately, thus neglecting the temporal smoothness. As a consequence they could not maintain a stable performance along time-series.

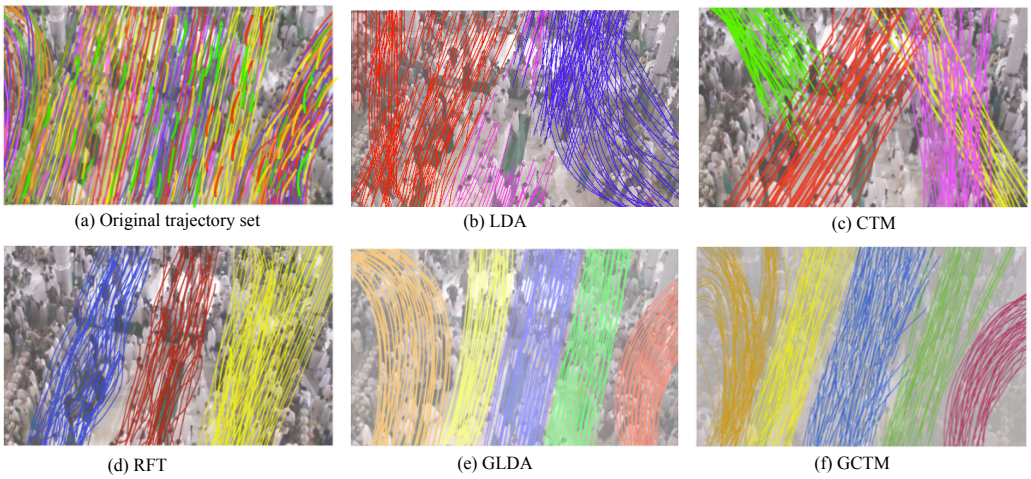
The GLDA achieved the highest performance in the CUHK dataset at K = 8, although the performance slightly dropped as the number of topics increased. This was caused by the independent assumption of the topic proportion generated from a Dirichlet in the GLDA. More topics would become correlated with increasing K, and the Dirichlet distribution would no longer be a good fit for such topic proportions. Construction of the STG helped the GLDA to perform much better than the other methods except the GCTM, which considered the correlations between topics during the learning process and thus had better ability to support larger numbers of topics.

Overall, GCTM performed better than GLDA. This was due to the limitation of LDA being incapable of modeling correlated topics, while CTM alleviated this limitation by introducing a logistic normal prior of topics to replace a Dirichlet prior and by using the covariance matrix of variables in the logistic normal model to capture correlations among topics. GLDA can be best applied to scenes in which each scene contains multiple topics, while the GCTM can be used to identify the relationships among the topics as well as topic detection.

Finally Figure 7 presents comparison of LDA, CTM, RFT, GLDA and GCTM with regard to the topic learning time. They included the processing time for feature extraction, codebook generation, topic learning and the final clustering. The figures show that the learning process of the proposed GLDA and GCT model were faster than LDA, CTM and RFT. Generating the LLC codes as low-level features, defining the STG between the trajectory segments and sup-

[1] Both LDA and CTM were implemented following the approach in [24].

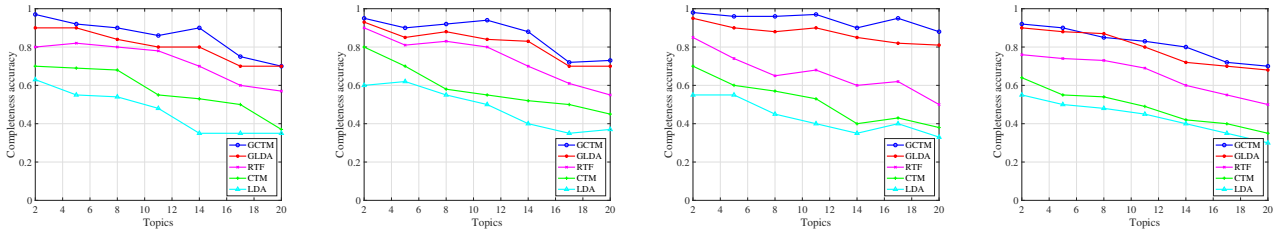
[2] We used the publicly available code from the authors' website [38].



(a) Original trajectory set (b) LDA (c) CTM

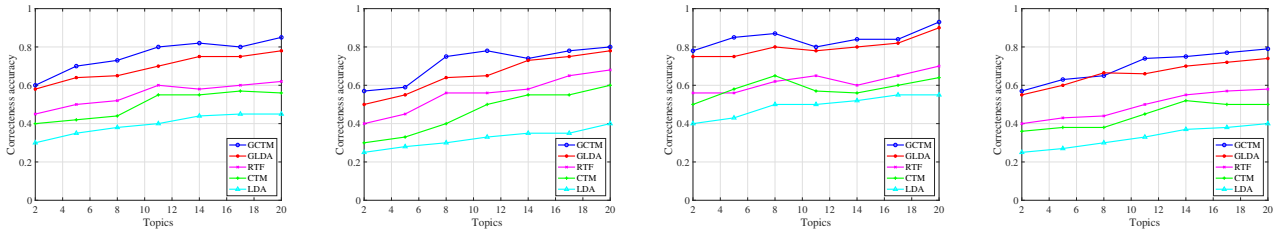
(d) RFT (e) GLDA (f) GCTM

Fig. 4 Comparison of trajectory clustering approaches using the Al-Masjid (S1) dataset: (a) original trajectory set, (b) LDA, (c) CTM, (d) RFT, (e) GLDA and (f) GCTM.



(a) Al-Masjid (S1) (b) Al-Masjid (S2) (c) Station (d) CUHK

Fig. 5 Completeness of trajectory clustering against the number of topics.



(a) Al-Masjid (S1) (b) Al-Masjid (S2) (c) Station (d) CUHK

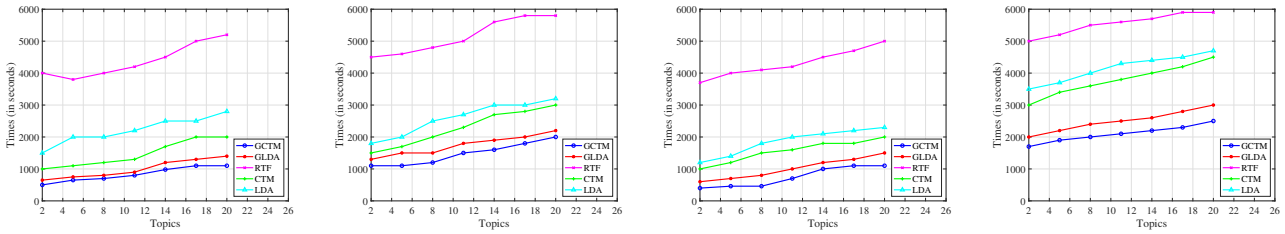
Fig. 6 Correctness of trajectory clustering against the number of topics.

porting the topic learning process with initial clusters helped to improve the computational aspects for topic modeling, while computing the scenes prior for RFT and tracking individuals with optical flow for LDA and CTM computationally more expensive.

5.2 The Effectiveness of the Low-level Features

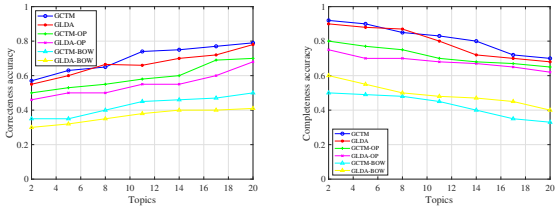
The low-level features (Section 3.1) were generated using the KLT tracker followed by the LLC algorithm to represent the extracted trajectories with a set of visual codes as low-level features. To demonstrate the effectiveness of this step, we compared its performance on the CUHK dataset with other models created in two approaches. In the first approach, low-level motion features were extracted through computing optical flow [8]. These motion features were then quantized into video words using the LLC. The second approach employed the KLT tracker to generate the trajectories which were then quantized into video words using the bag-of-words (BOW) algorithm [13]. We named the models in the first approach as GCTM-OP and GLDA-OP (OP for 'optical flow') and the models in the second approach as GCTM-BOW and GLDA-BOW.

Completeness and correctness for GCTM, GLDA, GCTM-OP, GLDA-OP, GCTM-BOW and GLDA-BOW are



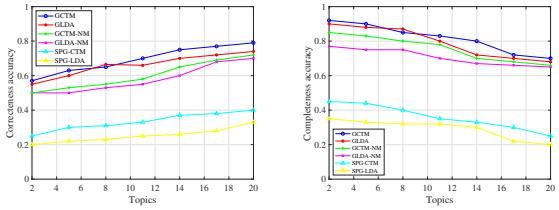
(a) Al-Masjid (S1) (b) Al-Masjid (S2) (c) Station (d) CUHK

Fig. 7 Comparison of the model learning time against the number of topics. A 2.6 GHz machine was used for computation.



(a) Correctness accuracy (b) Completeness accuracy

Fig. 8 Comparing the Completeness and Correctness of trajectory clustering using the presented models on the CUHK dataset with different low-level feature algorithms.



(a) Correctness accuracy (b) Completeness accuracy

Fig. 9 Comparing the Completeness and Correctness of trajectory clustering using the presented models on the CUHK dataset with different algorithms to define the initial clusters.

reported in Figure 8. As shown in the results, use of the KLT followed by the LLC for low-level features achieved better performances than the other methods. This was because the KLT tracklet are more conservative and less likely to drift in the crowd, while the optical flow was designed to detect local changes, not for recovering long-range motion patterns. Further, the neighbourhood graph in the LLC helped to handle the overlapping motion patterns, since each point was assigned to only one cluster. On the other hand, the BOW method utilized the spatial distance between the points to define the clusters. Replacing the LLC with the BOW had the largest impact in the models performances because it was the main step in creating the initial clusters used during the learning process. Consequently, replacing the KLT with optical flow followed by LLC achieved lower than the presented models but better than replacing the LLC with the BOW.

5.3 The Effectiveness of the Initial Clustering

To define the initial clusters (Section 3.2), the STG was generated to uncover the spatio-temporal relations between the trajectories and then projected to lower-dimensional space using the MDS [7]. To demonstrate the effectiveness of this step, we compared its performance on the CUHK dataset (the most challenging one) with other models created in two approaches. In the first approach, the STG was replaced in both the GCTM and the GLDA models with the spatial shortest path graph [29] that only considered the spatial information between the trajectories. While in the second approach, the manifold embedding was removed and the initial clusters were defined using the k-means algorithm on the STG. We named the models in the first approach as SPG-CTM and SPG-LDA (SPG for 'shortest path graph') and the models in the second approach as GCTM-NM and GLDA-NM (NM for 'no manifold').

Completeness and correctness for GCTM, GLDA, SPG-CTM, SPG-LDA, GCTM-NM and GLDA-NM are reported in Figure 9. It is clear that the result of the STG followed by the manifold embedding achieved better than the other methods. It showed the effectiveness of combing the STG with the manifold embedding techniques in utilizing the spatio-temporal correlation between trajectories in the learning process. The lowest performances were achieved by the models in the first approach with the shortest path graph.

6 Conclusions

In this paper we presented a graph-based topic models, GLDA and GCTM, for learning and clustering crowd motion from trajectory segments. Using a spatio-temporal graph and manifold-based clustering, the graph-based topic models could effectively capture the relations between trajectories, and learn discriminative motion patterns (topics) from crowd scenes. In the experiment they were compared with recent approaches, such as LDA, CTM and RFT, showing that GLDA and GCTM were faster to learn and more capable of modeling visual scenes for the trajectory cluster-

ing task. In particular the results showed that learnt topics by GLDA and GCTM were able to (1) separate different pathways at a fine scale with good accuracy, and to (2) capture the global structures of the scenes in long ranges, thus clearly interpreting crowd motion.

References

1. Alghamdi, M., Gotoh, Y.: Graph-based correlated topic model for motion patterns analysis in crowded scenes from tracklets. In: The British Machine Vision Conference (BMVC) (2018)
2. Alghamdi, M., Gotoh, Y.: Graph-based correlated topic model for trajectory clustering in crowded videos. In: The IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1029–1037 (2018)
3. Ali, S., Shah, M.: Floor fields for tracking in high density crowd scenes. In: ECCV, pp. 1–14 (2008)
4. Ali, Y., Zafar, B., Simsim, M.: Estimation of density levels in the holy mosque from a network of cameras. In: Traffic and Granular Flow, pp. 27–34 (2016)
5. Blei, D., Lafferty, J.: A correlated topic model of science. Annals of Applied Statistics pp. 17–35 (2007)
6. Blei, D., Ng, A., Jordan, M.: Latent dirichlet allocation. Journal of Machine Learning Research pp. 993–1022 (2003)
7. Borg, I., Groenen, P.: Modern multidimensional scaling: theory and applications. Springer (2005)
8. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: T. Pajdla, J. Matas (eds.) Computer Vision - ECCV, pp. 25–36. Springer Berlin Heidelberg (2004)
9. Burceanu, E., Leordeanu, M.: Learning a robust society of tracking parts using co-occurrence constraints. In: L. Leal-Taixé, S. Roth (eds.) Computer Vision – ECCV Workshops, pp. 162–178. Springer International Publishing, Cham (2019)
10. Chen, M., Wang, Q., Li, X.: Patch-based topic model for group detection. Science China Information Sciences **60**(11), 113101–113107 (2017)
11. Gariel, M., Srivastava, A.N., Feron, E.: Trajectory clustering and an application to airspace monitoring. IEEE Transactions on Intelligent Transportation Systems **12**(4), 1511–1524 (2011)
12. Kratz, L., Nishino, K.: Tracking pedestrians using local spatio-temporal motion patterns in extremely crowded scenes. IEEE Transactions on PAMI **34**(5), 987–1002 (2012)
13. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: The IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), vol. 2, pp. 2169–2178 (2006)
14. Leal-Taixé, L., Milan, A., Schindler, K., Cremers, D., Reid, I.D., Roth, S.: Tracking the trackers: An analysis of the state of the art in multiple object tracking. ArXiv **abs/1704.02781** (2017)
15. Lee, S., Baker, J., Song, J., Wetherbe, J.: An empirical comparison of four text mining methods. In: Proceedings of the Annual Hawaii International Conference on System Sciences, pp. 1 – 10 (2010)
16. Lin, W., Mi, Y., Wang, W., Wu, J., Wang, J., Mei, T.: A diffusion and clustering-based approach for finding coherent motions and understanding crowd scenes. IEEE Transactions on Image Processing **25**(4), 1674–1687 (2016)
17. Lu, W., Wei, X., Xing, W., Liu, W.: Trajectory-based motion pattern analysis of crowds. Neurocomputing **247**, 213–223 (2017)
18. Luo, W., Xing, J., Zhang, X., Zhao, X., Kim, T.: Multiple object tracking: a literature review. CoRR (2014)
19. Mao, Y., Zhong, H., Qi, H., Ping, P., Li, X.: An adaptive trajectory clustering method based on grid and density in mobile pattern analysis. In: Sensors (2017)
20. Moberts, B., Vilanova, A., van Wijk, J.: Evaluation of fiber clustering methods for diffusion tensor imaging. In: IEEE Visualization, pp. 65–72 (2005)
21. Pellegrini, S., Ess, A., Schindler, K., van Gool, L.: You'll never walk alone: Modeling social behavior for multi-target tracking. In: The IEEE 12th International Conference on Computer Vision, pp. 261–268 (2009)
22. Rabiee, H., Mousavi, H., Nabi, M., Ravanbakhsh, M.: Detection and localization of crowd behavior using a novel tracklet-based model. International Journal of Machine Learning and Cybernetics **9**(12), 1999–2010 (2018)
23. Raghavendra, R., Del Bue, A., Cristani, M., Murino, V.: Abnormal crowd behavior detection by social force optimization. In: A.A. Salah, B. Lepri (eds.) Human Behavior Understanding, pp. 134–145 (2011)
24. Rodriguez, M., Ali, S., Kanade, T.: Tracking in unstructured crowded scenes. In: ICCV, pp. 1389–1396 (2009)
25. Rodriguez, M., Sivic, J., Laptev, I., Audibert, J.Y.: Data-driven crowd analysis in videos. In: The International Conference on Computer Vision (ICCV), pp. 1235–1242 (2011)
26. Salti, S., Cavallaro, A., Di Stefano, L.: Adaptive appearance modeling for video tracking: Survey and evaluation. IEEE Transactions on Image Processing **21**(10), 4334–4348 (2012)
27. Serre, T., Wolf, L., Poggio, T.: Object recognition with features inspired by visual cortex. In: CVPR, pp. 994–1000 (2005)
28. Sharma, R., Guha, T.: A trajectory clustering approach to crowd flow segmentation in videos. In: ICIP, pp. 1200–1204 (2016)
29. Silva, V.D., Tenenbaum, J.B.: Global versus local methods in nonlinear dimensionality reduction. In: S. Becker, S. Thrun, K. Obermayer (eds.) Advances in Neural Information Processing Systems 15, pp. 721–728. MIT Press (2003)
30. Smeulders, A.W.M., Chu, D.M., Cucchiara, R., Calderara, S., Dehghan, A., Shah, M.: Visual tracking: An experimental survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(7), 1442–1468 (2014)
31. Tomasi, C., Kanade, T.: Detection and tracking of point features. Tech. rep., Carnegie Mellon University (1991)
32. van der Maaten, L., Postma, E., van den Herik, H.: Dimensionality reduction: a comparative review (2008)
33. Vedaldi, A., Fulkerson, B.: Vlfeat: an open and portable library of computer vision algorithms. In: International Conference on Multimedia, pp. 1469–1472 (2010)
34. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y.: Locality-constrained linear coding for image classification. In: CVPR, pp. 3360–3367 (2010)
35. Xu, H., Zhou, Y., Lin, W., Zha, H.: Unsupervised trajectory clustering via adaptive multi-kernel-based shrinkage. The IEEE International Conference on Computer Vision (ICCV) pp. 4328–4336 (2015)
36. Yuan, G., Xia, S., Zhang, L., Zhou, Y., Ji, C.: An efficient trajectory-clustering algorithm based on an index tree. Transactions of the Institute of Measurement and Control **34**, 850 – 861 (2012)
37. Zhou, B., Tang, X., Zhang, H., Wang, X.: Measuring crowd collectiveness. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(8), 1586–1599 (2014)
38. Zhou, B., Wang, X., Tang, X.: Random field topic model for semantic region analysis in crowded scenes from tracklets. In: CVPR, pp. 3441–3448 (2011)
39. Zou, J., Ye, Q., Cui, Y., Doermann, D., Jiao, J.: A belief based correlated topic model for trajectory clustering in crowded video scenes. In: ICPR, pp. 2543–2548 (2014)