CROWD MODELING USING SOCIAL NETWORKS

Rima Chaker      Imran N Junejo      Zaher Al Aghbari

University of Sharjah, U.A.E. 27272

ABSTRACT

In this work, we propose an unsupervised approach for detecting the anomalies in a crowd scene using social network model. Using a window-based approach, scene objects are first detected and tracked, and a spatio-temporal partitioning is constructed to produce a set of spatio-temporal cuboids that capture spatial and temporal features. A hierarchical social network is built to model the crowd behavior: the bottom-level models local behavior and the top level models the global. We perform anomaly detection and demonstrate the effectiveness of the proposed approach on a benchmark crowd analysis video sequences. Our results reveal that we outperform majority, if not all, the state-of-the-art methods.

Index Terms— Crowd Modeling, Social Network Model

1. INTRODUCTION

With the advent of technology and the increasing need for security round-the-clock, video surveillance has been receiving an increasing attention from the scientific community in recent years. One of the popular topics in intelligent video surveillance systems is the crowd analysis, i.e. how to automatically detect anomalies in public places or during public events. Early detection, or prediction, of abnormal behaviors occurring in surveillance scenes is of utmost significance. By alerting human operators, potential dangerous consequences can be reduced or prevented. However, the analysis of crowded scenes is a very challenging task, due to the fact that the analysis of human actions performed by individuals is still not a fully solved problem. Hence, crowd behavior analysis became an active research area in critical domains such as detection, or crowd monitoring and surveillance of riot.

Technically speaking, crowd behavior analysis comprises the following three main steps: (i) pre-processing (i.e. feature detection and tracking), (ii) motion information extraction and (iii) abnormal behavior modeling. Some researchers have attempted to track the objects throughout the scene. [1] use object tracking to detect unusual events in image sequences. Their method tracks each individual object in the scene and constructs models to determine abnormality in the objects’ behaviors. Similarly, [2] track subjects in high density crowded scenes, captured from a distance. They learn the direction of motion as prior information based on a force model floor fields. However, floor fields are chaotic in crowded scenes as tracking of each individual in pedestrian environments results in highly inconsistent trajectories, thus making the discrimination between usual and unusual events extremely difficult. Finally, this approach is considered appropriate only in scenes containing only a few objects, as it is difficult to reliably segment and track each individual in a crowded environment.

Tracking in a crowded scene is a daunting task, hence researchers resort to optical flow or tracklets: [3] explored the socio-psychological concept social force in combination with optical flow to compute interaction forces that are later combined with Latent Dirichlet Allocation to model normal behaviors and detect abnormal ones. This method is further extended in [4] using Particle Swarm Optimization, in addition to social force model, to optimize the computed interaction force and thus detect global abnormal activities. [5] draw inspiration from the existence of Coherent Structures in fluid dynamics for segmenting dominant crowd flows and flow instability detection. Perhaps a more intuitive approach is to find interest points and track them over time. [6] analyzed motion patterns by clustering the extracted tracklets in a crowded scenes. Despite the many different representations of video events, many of the existing works ignored the importance of “contextual” anomaly in the field of crowd analysis. Contextual anomaly arises when an individual apparently exhibits a behavior similar to the others but it is anomalous in a specific context (e.g. neighborhood). [7] focus on detecting contextual anomalies in the context of neighborhood motion based on statistical analysis of detected blobs. [8] detect subtle context-dependent behavioral anomalies based upon contextual information. The strength of their contextual features is demonstrated by social and scene contexts. Social context prevent self-justifying groups and propagate anomalies in social network. Scene context improves the detection of subtly abnormal behaviors.
Beside the motion information, other works included important object features such as appearance or size. [9] handled this restriction by applying Mixture of Dynamic Textures (MDT) to jointly model the appearance and dynamics of crowded scenes. Their approach is more reliable in anomaly localization since it investigates both temporal and spatial abnormalities. In [10] a sparse reconstruction cost is proposed to detect the presence of anomalies in crowded scenes. They adopted the local spatio-temporal patches to construct the normal dictionary that measures the abnormality of a test sample.

In this work, the proposed method is based on the social network model. In this regard, [11] explored the study of crowd social grouping to solve the multi-person data-association-based tracking (DAT) problem. Similarly, [12] study of social behavior among animals over prisons. In addition, their method requires the identification of group leaders. However they addressed the problem in closed-world surveillance environments i.e. prisons. In addition, their method requires the identification of individuals with respect to a pre-defined watch-list of faces.

2. SCENE MODELING WITH SOCIAL NETWORKS

This section describes the proposed method, which contains the following steps: (i) extract tracklets of the human subjects, (ii) Partition the current time window \( \Omega_t \) into spatio-temporal cuboids \( \mathcal{C}_j \), (iii) construct local and global social networks, and (iv) perform anomaly detection. KLT tracker [14] is used to obtain the tracklets of the objects in the scene.

In order to group tracklets that exhibit similar behavior, especially for our application, we focus on selecting the features that account for the (i) direction and magnitude of the motion, and (ii) distance between the moving objects. Thus, we use the following measures:

- **Cosine Similarity**: Cosine similarity measures the cosine of the angle between two tracklets and the value is in the range \([0; 1]\). Let \( \phi_{\tau_u}, \phi_{\tau_v} \) denote the dominant directions of tracklets \( \tau_u \) and \( \tau_v \) respectively, the cosine similarity is defined as [15]:

  \[
  s_{\phi}^{\tau_u, \tau_v} = \left( \frac{\phi_{\tau_u} \cdot \phi_{\tau_v}}{\|\phi_{\tau_u}\| \|\phi_{\tau_v}\|} + 1 \right) \times \frac{1}{2}
  \]

- **Magnitude Similarity**: Let \( \rho_{\tau_u}, \rho_{\tau_v} \) denote the magnitudes of tracklets \( \tau_u \) and \( \tau_v \) respectively, the magnitude similarity is defined as [15]:

  \[
  s_{\rho}^{\tau_u, \tau_v} = \frac{1}{\max(\rho_{\tau_u}, \rho_{\tau_v})} |\rho_{\tau_u} - \rho_{\tau_v}|\]

Combining the above two similarities produces:

\[
 s_{\phi\rho}^{\tau_u, \tau_v} = \alpha s_{\phi}^{\tau_u, \tau_v} + (1 - \alpha) s_{\rho}^{\tau_u, \tau_v}
\]

with \(0 \leq \alpha \leq 1\) is the learning parameter.

- **Velocity Similarity**: The similarity in motion characteristics is denoted by \( s_v^{\tau_u, \tau_v} \) and is computed using the Dynamic Time Warping (DTW) algorithm, where the distance measure used is:

  \[
  d(v_i, v_j) = \sqrt{d_{vx}^2 + d_{vy}^2} ,
  \]

where \(d_{vx} \) and \(d_{vy}\) represents velocity distance between the two tracklets along the x and y axis, respectively.

**Definition 1 (Social Similarity Measure)**: The social similarity measure between \( \tau_u \) and \( \tau_v \) is defined as:

\[
 s_{\phi\rho\nu}^{\tau_u, \tau_v} = \beta s_{\phi\rho}^{\tau_u, \tau_v} + (1 - \beta) s_v^{\tau_u, \tau_v}
\]

with \(0 \leq \beta \leq 1\) is the learning parameter.

2.1. Building Social Networks

Social network is a social structure represented as a graph in which nodes represent objects and edges represent social interactions between pedestrians. The social interaction weights are based on our social similarity weight measure eq (2). Once the tracklets are obtained from the scene, we apply the spatio-temporal algorithm that produces a dense number of cuboids \( \mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_{N_{cub}} \times N_{cub} \) to the time windows \( \Omega_1, \Omega_2, \ldots, \Omega_{N_t} \) (cf. Fig. 1). Within each cuboid in each time window, we do the following:

**(1) Closeness centrality** On a graph, a geodesic between two nodes is a path connecting the nodes with the smallest number of edges. The classical definition of the closeness centrality is (the inverse of) the average distance to all other nodes [16]. Since similar behaving tracklets need to be spatially close to each other, in addition to the social similarity measure, i.e. eq (2), we use the closeness centrality among connected tracklet nodes for pruning only: if similar tracklet nodes are spatially distant (threshold \( t_d \)), their connecting edge is deleted. This is then followed by applying the connected component
algorithm to the whole network. The idea is to identify the different dynamics of the scene, represented by the clusters in the network. We call each of these clusters obtained at this stage as local social network (LSN).

**Local Social Network:** A centroid is used to represent each LSN, denoted by $\text{Cent}_i^{SN}$. It consists of the mean vector of its tracklets' features, which are spatial $<x,y>$, direction $\phi$, magnitude $\rho$, and velocity $v$. These LSN capture the local dynamics of each $\varsigma_i$.

**Global Social Network (GSN):** Once we obtain $LSN_i$ from the previous step, clustering is employed to have a coarser view of the scene. Our measure of the cluster centroid similarity is defined by equation (2). If similarity of two comparable social network’s centroid is sufficiently high, i.e. greater than a user-defined threshold $t_f$, then the networks are merged together into one bigger network, otherwise they are reported as disjoint networks. For this, we perform Hierarchical Agglomerative Clustering [17]. Fig.1 illustrates the framework of our hierarchical clustering algorithm. Thus the merging is based on the social similarity measure as defined above - between every two LSN components’ centroid, i.e. $\text{Cent}_i^{SN}$ and $\text{Cent}_j^{SN}$ of $LSN_i$ and $LSN_j$, respectively. If the similarity measure between $LSN_i$ and $LSN_j$ is above $t_f$, they merge to form: $GSN_{ij} = LSN_i \cup LSN_j$, and a new Cent for this is computed as well.

This bottom-up approach aims to merge similar LSNs from different cuboids towards discovering the global social network $GSN_j$ within time window $\Omega_k$, thus capturing the dynamics of the whole window $\Omega_i$ (cf. Fig 2).

### 2.2. Anomaly Detection

Our social similarity measure, in addition to the size of social network, is considered an essential step for identifying abnormality. For each cuboid $\varsigma_j$, to identify which local social network component(s) $LSN_i^{\varsigma_j}$ is anomalous, we use:

$$LA_i^{\varsigma_j} = \frac{LSN_i^{\varsigma_j}}{\text{maxsize}(LSN_i^{\varsigma_j})}, 1 \leq i \leq n \text{ and } i \neq h$$

Thus, if the relative size of the tested local social network component is less than $t_h\%$ the size of the densest local social network component in the neighborhood, it is classified as an anomalous local social network component.

Global anomalies are detected in the similar way, using:

$$GA_i^{\Omega_k} = \frac{GSN_{h_i}^{\Omega_k}}{\text{maxsize}(GSN_{j_i}^{\Omega_k})}, 1 \leq j \leq n \text{ and } j \neq h$$

if the relative size of the tested $GSN_{h_i}^{\Omega_k}$ is less than $t_h\%$ the size of the densest GSN component in the neighborhood $\text{maxsize}(GSN_{j_i}^{\Omega_k})$, it is classified as an anomalous. In summary, the method starts by determining the size of each social network component based on the number of its tracklets members. Then it determines the largest or densest social network in the neighborhood. The algorithm, hence, uses a threshold $t_a$ to differentiate each non-densest social network as normal or abnormal component. If the relative size of each tested social network component is less than $t_h\%$ the size of the densest social network component in the neighborhood then it is classified as an anomalous social network component.

### 3. EXPERIMENTS & RESULTS

We conducted an extensive set of experiments on the following crowd dataset: the UCSD pedestrian dataset. The dataset contains two sequences: UCSD Ped1, containing groups of people walking towards and away from the camera with some amount of perspective distortion, and UCSD Ped2, containing groups of people walking in parallel to the camera plane (cf. Fig. 3). Commonly occurring anomalies include small golf carts in the scene, skaters, bikes, and people in wheelchairs. Each scene
was divided into clips of about 200 frames, and resolution of 158 x 238 for UCSD Ped1 and resolution 360 x 240 for UCSD Ped2. Our proposed method run all the experiments on a PC computer with an Intel(R) Core(TM) i5–2400 3.10GHz CPU and 4GB RAM under the MATLAB implementation. For both sequences, the search window size for tracking is set to 16 x 16 pixels in the Lucas-Kanade based methods, with three pyramid levels, around 1000 foreground features and 4 as the minimum distance imposed between corner features. For the construction of feature vectors, we partition each 50 frames time window into 8 x 8 spatio-temporal cuboids. The values for $\alpha$ and $\beta$ are chosen experimentally to be 0.8 and 0.4, respectively; and $t_\% = 50\%$.

We use the **frame-level criterion**: a frame is considered an anomaly if it contains at least one abnormal pixel, and denoted as positive. For the LSN, we measure the frame accuracy. For the global evaluation, we use the Receiver Operating Characteristic (ROC) curve, which is based on True Positive Rate (TPR) and False Positive Rate (FPR) [18]. **Area under the curve** (AUC) is computed from the ROC curve to compare our method to the other methods, as shown in Fig. 4, and compared in Table 1. Rate of Detection [9] is shown in Table 2.

It is clear that our method clearly outperforms the state-of-the-art methods.

### 4. CONCLUSION

We present a general method that covers both local and global anomalous events without the need for background/foreground segmentation and individual tracking. We performed learning and tested on standard dataset and compared our results with the state-of-the-art. The main advantages of our method are that it learns the dominant behavior in an unsupervised manner while simultaneously detecting anomalous patterns. Such characteristic of a visual surveillance system operates better in an unconstrained environment.

### 5. REFERENCES


