University Library



A gateway to Melbourne's research publications

Minerva Access is the Institutional Repository of The University of Melbourne

Author/s:
Kumar, D;Bezdek, JC;Rajasegarar, S;Leckie, C;Palaniswami, M

Title:
A visual-numeric approach to clustering and anomaly detection for trajectory data

Date:
2017-03-01

Citation:
Kumar, D., Bezdek, J. C., Rajasegarar, S., Leckie, C. & Palaniswami, M. (2017). A visual-numeric approach to clustering and anomaly detection for trajectory data. Visual Computer, 33 (3), pp.265-281. https://doi.org/10.1007/s00371-015-1192-x.

Persistent Link:
https://hdl.handle.net/11343/283282

The Visual Computer manuscript No.
(will be inserted by the editor)

A Visual-Numeric Approach to Clustering and Anomaly Detection for Trajectory Data

Dheeraj Kumar · James C. Bezdek · Sutharshan Rajasegarar · Christopher Leckie · Marimuthu Palaniswami

Received: date / Accepted: date

Abstract This paper proposes a novel application of *Visual Assessment of Tendency* (VAT) based hierarchical clustering algorithms (VAT, iVAT, and clusiVAT) for trajectory analysis. We introduce a new clustering based anomaly detection framework named iVAT+ and clusiVAT+ and use it for trajectory anomaly detection. This approach is based on partitioning the VAT generated *Minimum Spanning Tree* (MST) based on an efficient thresholding scheme. The trajectories are classified as normal or anomalous based on the number of paths in the clusters. On synthetic datasets with fixed and variable numbers of clusters and anomalies, we achieve 98% classification accuracy. Our two-stage clusiVAT method is applied to 26,039 trajectories of vehicles and pedestrians from a parking lot scene from the real life MIT trajectories dataset. The first stage clusters the trajectories ignoring directionality. The second stage divides the clusters obtained from the first stage by considering trajectory direction. We show that our novel two-stage clusiVAT approach can produce natural and informative trajectory clusters on this real life dataset while finding representative anomalies.

Index Terms - Trajectory clustering, Anomaly detection, clusiVAT hierarchical clustering, MIT trajectory dataset

Dheeraj Kumar · Marimuthu Palaniswami
Department of Electrical and Electronic Engineering,
The University of Melbourne, Australia.
E-mail: {dheerajk@student., palani@} unimelb.edu.au

James C. Bezdek · Sutharshan Rajasegarar · Christopher Leckie
Department of Computing and Information Systems,
The University of Melbourne, Australia.
National ICT Australia, Victoria.
E-mail: {jbezdek, caleckie, sraja}@unimelb.edu.au

1 Introduction

Security and surveillance in public environments have been topics of active research for the past few decades. With the increased popularity and easy deployment of *closed circuit television* (CCTV) cameras, digital surveillance has become a common practice. Videos acquired by surveillance cameras require continuous monitoring by a human agent for suspicious activity detection, which is tedious and expensive. Consequently, a recent focus of research has been on automatic event detection from video sequences [4,9,11,19,24,27,41,44]. Much of the recent work concentrates on the recognition of simple activities like running, waving and jumping using high resolution videos. However, for the far-field, low-resolution video obtained from CCTV cameras, detailed information for activity recognition is not available. The only information that this type of video system can reliably extract is a noisy trajectory [2, 8, 26, 31, 36]. While supervised learning can be used for explicit event recognition from extracted trajectories, it is not appropriate for exploratory analysis of huge volumes of video data from CCTV cameras in public environments because (i) explicit knowledge of the event to be identified is not available; and (ii) the events of interest are not labeled. A more appropriate approach for analysis of large volumes of low resolution video data is unsupervised learning - in particular, clustering of trajectories into different activities and then using the clusters to identify abnormal trajectories.

The goal of this work is to partition sets of trajectories into various clusters, so that trajectories within a cluster are close to each other in the sense of some distance measure (either Euclidean or *Dynamic Time Warping* (DTW)) as compared to trajectories in other clusters. Trajectories belonging to clusters that have

very few trajectories and are far apart from the other clusters are defined as anomalous. Clustering algorithms invariably require k, the number of clusters to seek, as a runtime input. The VAT [5], iVAT [14] and clusiVAT [20, 21] algorithms enable users to make an estimate of k from a visual image of the (reordered) data. However, visual interpretation the image may vary from one user to another. A primary motivation for this paper is to eliminate the subjectivity involved in estimating k, by developing a method that automatically detects k in trajectory data. Once k is estimated, we demonstrate how to find the clusters and anomalies in the data. Our main contributions in this paper are as follows:

- We develop two novel trajectory anomaly detection approaches named iVAT+ and clusiVAT+ for small and big data cases respectively, and apply them on publicly available synthetic trajectory datasets. Our experiments show that the new techniques achieve better accuracy than several popular methods that have previously been reported in the literature.
- We propose and experiment on a novel two-stage clusiVAT algorithm on a real life MIT dataset of trajectories of vehicles and pedestrians in a parking lot scene.
  - The first stage discards the directions of trajectories in order to group together trajectories that follow similar paths (possibly in different directions) using clusiVAT.
  - In the second stage, the clusters obtained from the first stage are re-clustered using their directionality to separate trajectories that are heading in opposite directions.
  - We then detect anomalous trajectories using the clusiVAT+ algorithm.

The rest of the paper is organized as follows. Section 2 provides a literature review of trajectory clustering and anomalous trajectory detection methods. Section 3 describes the distance measure between pairs of trajectories needed for clustering and anomaly detection using the iVAT and clusiVAT algorithms, which are described in Section 4. Our clustering based anomaly detection frameworks, iVAT+ and clusiVAT+ are discussed in Section 5 and their time complexity is described in Section 6. Quantitative experiments on two synthetic and one real life trajectory dataset are given in Section 7 before summarizing our results in Section 8.

2 Related Work

Activity analysis has long been one of the principal areas of research in surveillance. Previous approaches include: motion patterns based on histogram of pixel changes [4], local optical flow combined with probabilistic *principal component analysis* (PCA) [19], optical flow with interaction features (e.g., social force models) [11, 27], and histograms of optical flow [9, 44]. Anomalous event detection in video [10, 12, 37, 45] is a challenging problem because it is difficult to define anomalies explicitly. The system needs to identify an anomaly when it appears, despite the fact that it has never occurred before. A better approach is to characterize normal events first (as they follow some regular rules) and then treat non-normal events as anomalies. Using this approach Li et al. [24] provide a statistical treatment of anomaly detection, where events of low probability with respect to a probabilistic model of normal behaviour are declared as anomalous.

In many cases, however, a priori knowledge of regular rules is lacking and no training data for normal video events are available. Therefore, there is a need for an unsupervised approach to automatically mine these rules directly from unlabeled data. Clustering-based approaches have recently been investigated to address this problem [16, 17, 33]. These approaches are based on the fact that normal events appear frequently and dominate the data, while anomalies differ from the normal case and appear rarely. Events that are clustered into dominant (e.g., large) groups can be identified as normal, representing the regular patterns. Those that cannot be explained by the regular patterns (e.g., outliers distant from all cluster centers) are anomalous. These approaches extract the track of moving objects from a video and then model their trajectories, which separates activities performed by different objects having different tracks.

Although individual object detection has improved significantly over the years, crowded scenes remain challenging for detection and tracking tasks due to factors such as occlusions, high person densities and significant variations in the appearance of people. Recent work in the area of detecting and tracking multiple people in high density crowd scenes include [2, 6, 26, 35, 36]. These methods use a motion-based approach to identify and track multiple objects, which can be ineffective in the case of dynamically changing background scenes. The studies in [8, 31] propose various background subtraction methods to minimize the effect of significant stochastic motion of the background.

Jiang et al. [17] track all the moving objects in a video to provide three different levels of spatiotemporal context, i.e., a point anomaly of a video object, a sequential anomaly of an object trajectory, and a co-occurrence anomaly of multiple video objects using a hierarchical data mining approach. Arandjelović

[3] uses a clustering-based algorithm for estimating a scene specific low-level motion model termed as "tracklet primitive." This is followed by motion novelty inference from a non-hierarchial ensemble of Markov chains as a means of capturing behavioural characteristics at different scales. Saligrama et al. [39] describes a family of unsupervised approaches to video anomaly detection based on statistical activity analysis. Saleemi et al. [38] proposed a stochastic, nonparametric method for modeling scene tracks using a kernel density estimation based multivariate nonparametric probability density function of spatiotemporal variables. Morris and Trivedi [29] evaluated different similarity measures and clustering methodologies that have appeared in the literature, and catalogued their strengths and weaknesses when utilized for the trajectory learning problem.

Piciarelli et al. [32, 33] used a one-class Support Vector Machine (SVM) for anomalous trajectory detection in a synthetic dataset with 1 to 10 clusters having 100 trajectories each, to which 1 to 10 anomalous trajectories were added. The authors in [32, 33] used geometric considerations in the SVM feature space for automatic detection of the outliers. Laxhammar and Falkman [22] proposed two parameter-free dissimilarity measures based on Hausdorff distance for comparing multi-dimensional trajectories of arbitrary length for on-line learning and anomaly detection. The Euclidean distance is perhaps the simplest and most intuitive distance measure, but requires that trajectories are properly aligned and of equal length. DTW has been proposed for relaxing alignment constraints [28]. Other approaches reduce trajectories to a fixed low-dimensional representation using various transforms [30]. Abstract and hierarchical representations have also been proposed where trajectories are composed of a number of basic movement patterns, such as motifs [25].

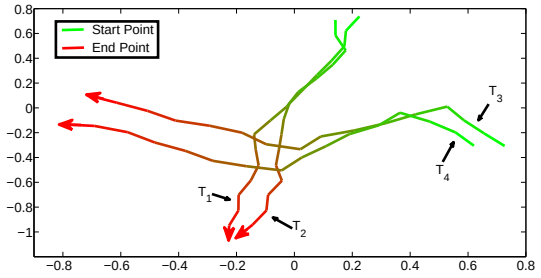
Wang et al. [43] used an unsupervised nonparametric Bayesian model, which they called Dual Hierarchical Dirichlet Processes (Dual-HDP) for trajectory clustering. Trajectories with low likelihoods are declared as anomalies. The experiments were performed on a real dataset of 40,453 trajectories in a parking lot collected over one week (MIT Trajectory Dataset [42]). Brun et al. [7] used string kernels to define a distance measure between two trajectories, and processed them with a novel kernelized clustering algorithm. Abnormal trajectories in the MIT trajectory data were classified in [7] based on the statistical properties of the clusters. Next we describe two methods for measuring the distance between a pair of trajectories as a preparatory step for clustering and anomaly detection.

3 Distance measures for trajectory data

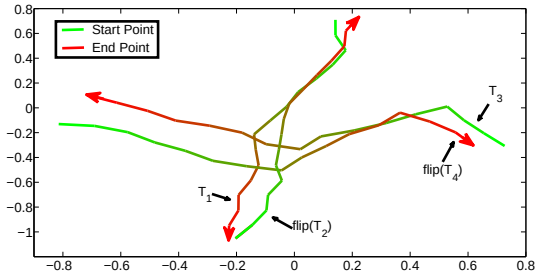
Trajectories are usually represented by the points sampled along the arc traversed by an object at equal time intervals. The distance measure between two trajectories depends on incremental path lengths, which may or may not be equal, and also on the direction of the moving object. For two trajectories in a 2-D plane represented by equal numbers of sample points m, $T_i = \{t_i^1, t_i^2, ..., t_i^m\}$ and $T_j = \{t_j^1, t_j^2, ..., t_j^m\}$, where $t_i^k : 1 \leq k \leq m$ represents the tuple $(t_{i_x}^k, t_{i_x}^k)$, the Euclidean distance is given by

$$d_{ij} = ||T_i - T_j||_E = \sqrt{\sum_{k=1}^{m}((t_{i_x}^k - t_{j_x}^k)^2 + (t_{i_y}^k - t_{j_y}^k)^2)}. \quad (1)$$

The directionality of trajectories can result in misleading distances between them, which in turn may cause incorrect clustering results, as explained by the following example. Fig. 1 shows four trajectories represented by $m = 16$ sampled points along the travelled path. The path is color coded to show the direction of the trajectory. Green represents the starting point and red denotes the ending point, also shown by the arrow at the end. View (a) of Fig. 1 shows trajectories T_1 and



(a) Parallel trajectories



(b) Anti-parallel trajectories

Fig. 1: The Euclidean distance between pairs of trajectories

T_2 going from up to down (belonging to a first cluster), and trajectories T_3 and T_4 going from right to left (belonging to a second cluster). The (Euclidean) distance matrix of these four trajectories is given in Table 1. The (intra-cluster) Euclidean distance between T_1 and T_2 is 0.08, and between T_3 and T_4 is 0.53, which is much smaller than the inter-cluster distances ($d(T_1, T_3) = 7.76$, $d(T_1, T_4) = 6.36$, $d(T_2, T_3) = 8.02$, and $d(T_2, T_4) = 6.79$). This distance matrix validates our hypothesis that $\{T_1, T_2\}$ belong to one cluster and $\{T_3, T_4\}$ belong to another cluster.

Table 1: (Euclidean) Distance matrix for T_1, T_2, T_3, and T_4

Distance	T_1	T_2	T_3	T_4
T_1	0	0.08	7.76	6.36
T_2	0.08	0	8.02	6.79
T_3	7.76	8.02	0	0.53
T_4	6.36	6.79	0.53	0

If we reverse the direction of the trajectories T_2 and T_4 and denote them as flip(T_2) and flip(T_4) respectively as shown in Fig. 1(b), the new distance matrix is given in Table 2. In the new distance matrix $d(T_1, flip(T_2)) = 20.34$, which is higher than $d(T_1, T_3) = 7.76$ and $d(T_1, flip(T_4)) = 10.84$, suggesting that flip($T_2$) is farther than T_3 and flip(T_4) to T_1. Similarly, flip(T_4) is farther to T_3 ($d(T_3, flip(T_4)) = 13.66$) than it is to T_1 ($d(T_1, flip(T_4)) = 10.84$) and flip($T_2$) ($d(flip(T_2), flip(T_4)) = 6.79$), producing an erroneous implication suggesting that T_1 and T_3 belong to one cluster and flip(T_2) and flip(T_4) belong to another cluster.

Table 2: (Euclidean) Distance matrix for T_1, flip(T_2), T_3, and flip(T_4)

Distance	T_1	flip(T_2)	T_3	flip(T_4)
T_1	0	20.34	7.76	10.84
flip(T_2)	20.34	0	9.52	6.79
T_3	7.76	9.52	0	13.66
flip(T_4)	10.84	6.79	13.66	0

This problem of having an erroneous distance measure due to the directionality of the trajectories can be easily solved by reversing one trajectory (so that the starting point becomes the ending point and vice versa) and taking the minimum value of the distance between the first trajectory and the second trajectory, and the first trajectory and the reversed second trajectory. We denote this measure as nondirectional Euclidean distance, which is given as

$$\text{nondirectional}(||T_i - T_j||_E) = min(||T_i - T_j||_E, ||\text{flip}(T_i) - T_j||_E), \quad (2)$$

where $\text{flip}(T_i) = \{t_i^m, t_i^{m-1}, ..., t_i^1\}$ reverses the coordinate order of T_i, as shown in Fig. 1(b).

Although this solves the problem of directionality, it groups trajectories with opposite directions into the same cluster, which may not be desirable for some applications. To solve this problem we propose a novel two-stage clustering procedure as explained below. In the first step we use nondirectional Euclidean distance to cluster the trajectories, so that trajectories which have opposite starting and finishing points, but follow similar paths, are clustered into the same group. In the second stage, we use directional Euclidean distance for each cluster of trajectories obtained in the first step to separate the trajectories going in opposite directions.

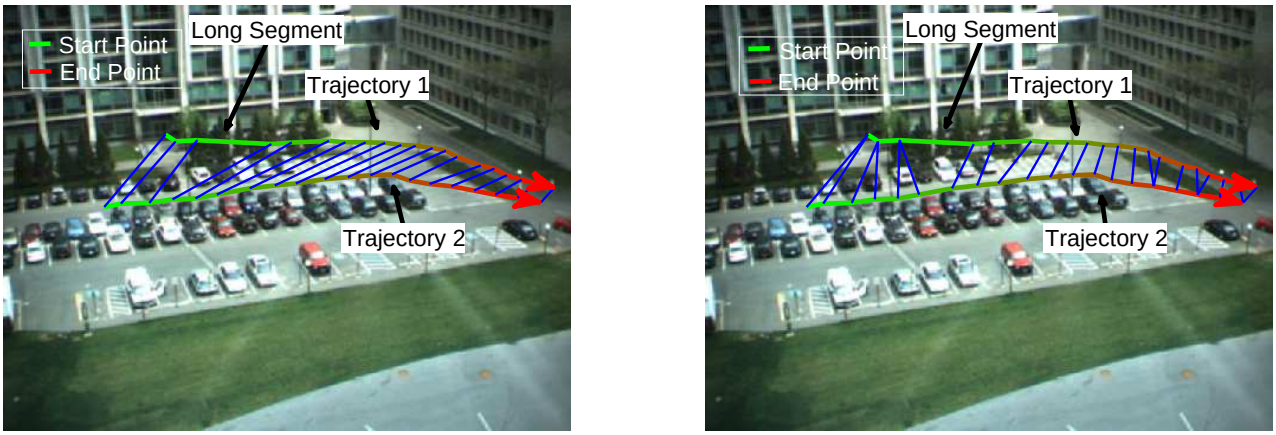
For trajectories having different numbers of sample points, Euclidean distance is not defined. Also for trajectories sampled at different time intervals, Euclidean distances can be misleading. In such cases DTW [34] (Algorithm 1) is a much better distance function [28].

Algorithm 1: DTW

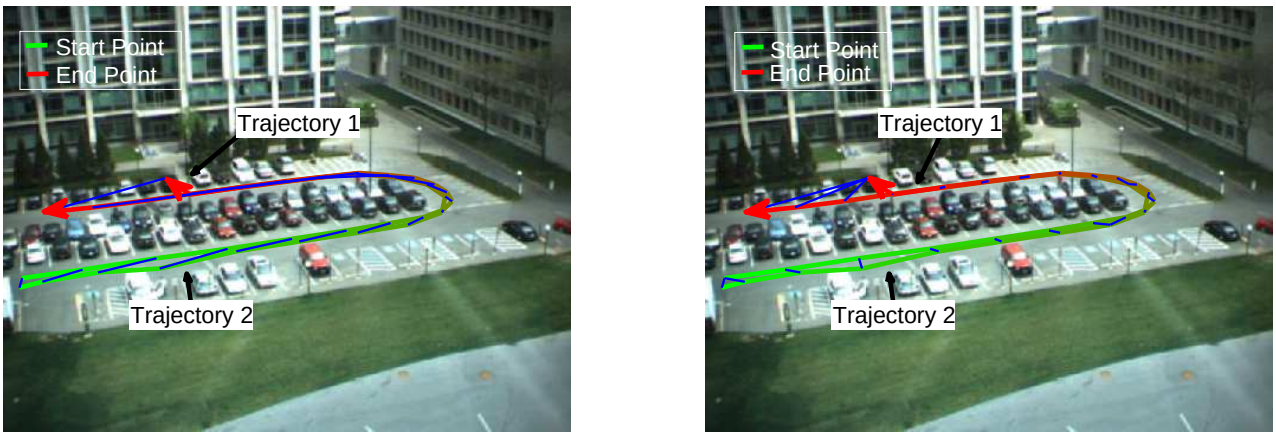
Input : $T_i = \{t_i^1, t_i^2, ..., t_i^m\}$ − Trajectory i having m sample points
$T_j = \{t_j^1, t_j^2, ..., t_j^l\}$ − Trajectory j having l sample points
w − Window parameter
Output: DTW(T_i, T_j) − DTW distance

```
for k ← 1 to m + 1 do
    A_{k,1} = ∞
end
for k ← 1 to l + 1 do
    A_{1,k} = ∞
end
A_{1,1} = 0
for k ← 2 to m + 1 do
    for p ← max(k − w, 2) to min(k + w, l + 1) do
        cost = ||t_i^k − t_j^p||_E
        A_{k,p} = cost + min(A_{k−1,p}, A_{k,p−1}, A_{k−1,p−1})
    end
end
DTW(T_i, T_j) = A_{m+1,l+1}
```

DTW is an algorithm for measuring similarity between two temporal sequences T_i and T_j, which may vary in time or speed. The window parameter w adds a locality constraint to DTW. That is, we require that if $T_i^a, 1 \leq a \leq m$ is matched with $T_j^b, 1 \leq b \leq l$, then $|a - b| \leq w$. When $w = 0$, the DTW distance is same as Euclidean distance. As we increase w, the DTW algorithm has some flexibility in matching the corresponding points on two trajectories, but if w is too high, the



(a) Euclidean distance (left) ($d_{ij} = 64.5$) & DTW distance (right) ($d_{ij} = 33.6$)



(b) Euclidean distance (left) ($d_{ij} = 57.3$) & DTW distance (right) ($d_{ij} = 17.7$)

Fig. 2: Euclidean vs DTW distances for two pairs of trajectories

warping is too much and the DTW distance is not representative of the actual distance between the trajectories. In our experiments, the trajectory length varies from 16 (for the synthetic trajectory dataset) to 20 (for the MIT trajectory dataset), hence we set w to be around 15% of the trajectory length, giving $w = 3$ for all the demonstrations and experiments in this paper. Setting w to a much higher value than this would decrease the distance between non-similar trajectories producing erroneous results. As a general rule, w should be in the range of 10-20% of the trajectory length. Fig. 2 compares the Euclidean and DTW distances between two pairs of trajectories taken from the MIT trajectories dataset [42]. Both pairs of trajectories change color from green (starting point) to red (ending point). The blue lines represent the matched points of the two trajectories. View (a) of Fig. 2 shows two trajectories having non-uniform spatial sampling, one of which has a relatively long spatial segment (the "Long Segment"). If we use Euclidean distance, this causes incorrect matching of the corresponding points, resulting in a large distance value of 64.5, whereas DTW aligns the corresponding points of the two trajectories, obtaining a smaller distance of 33.6. Similarly Fig. 2 (b) shows two trajectories that almost coincide for about 80% of the trajectory but differ for the remainder. Euclidean distance overestimates the actual distance with a value of 57.3, whereas DTW aligns the corresponding points on the trajectories, resulting in the lower distance of 17.7.

We define the nondirectional DTW distance between two trajectories T_i, and T_j as

$$\text{nondirectional_DTW}(T_i, T_j) = min(\text{DTW}(T_i, T_j), \text{DTW}(\text{flip}(T_i), T_j)). \quad (3)$$

Next we describe the VAT family of algorithms used for clustering and anomaly detection in trajectory data.

4 VAT, iVAT and clusiVAT Algorithms

Reordered dissimilarity image (RDI) algorithms like the *visual assessment of cluster tendency* (VAT) and *improved VAT* (iVAT) produce visual images (sometimes called cluster heat maps) of a given input data dissimilarity matrix. VAT [5] reorders the input distance matrix D to obtain D^* using a modified version of Prim's algorithm. The image $I(D^*)$, when displayed as a gray-scale image, shows possible clusters as dark blocks along the diagonal. The other outputs of the VAT algorithm are P, which represents the VAT reordering indices of the distance matrix D, i.e., $D^* = D_{P,P}$ and d, the array representing the length of edges of the VAT generated MST.

Although VAT can provide a useful estimate of the number of clusters in a dataset, VAT images are often inconclusive. A second algorithm called *improved VAT* (iVAT) was proposed in [14]. iVAT provides better images by replacing the input distances in the VAT reordered distance matrix $D^* = [d^*_{ij}]$ by geodesic distances $D'^* = [d'^*_{ij}]$, given by

$$d'^*_{ij} = \min_{p \in P_{ij}} \max_{1 \le h \le |p|} D^*_{p[h]p[h+1]}, \quad (4)$$

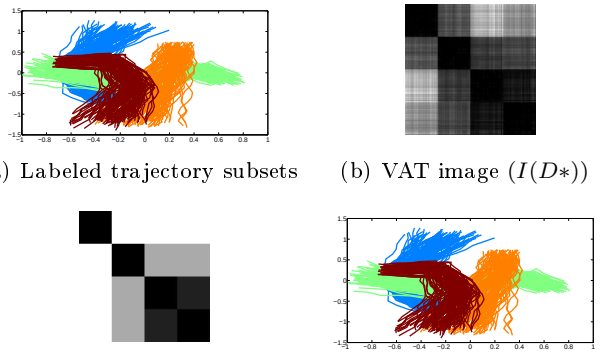
where P_{ij} is the set of all paths from trajectory i (T_i) to trajectory j (T_j) in the VAT generated *minimum spanning tree* (MST). iVAT has a time complexity of $O(n^2)$ and the theory that connects the *single linkage* (SL) clustering algorithm to VAT also holds for iVAT, which preserves the VAT order.

VAT and iVAT are limited by screen resolution and software limits to input datasets where the number of objects N is of the order of $10,000$. To overcome this limitation, *scalable-VAT* (sVAT) was introduced in [13], which works by sampling a large dataset $T = \{T_1, T_2, ..., T_N\}$ having $N > 10,000$ trajectories. sVAT applies VAT to a small $n \times n$ distance matrix D_n of the samples, where n is a "VAT-sized" fraction of N. siVAT is just like sVAT, except it uses iVAT after the sampling step.

The MST built using Prim's algorithm in VAT and iVAT provides an array that represents the order of insertion of edges into the MST, which is then used in the reordering operation. Let us assume that the iVAT image suggests the presence of k clusters in the dataset D_n. Having this estimate, we cut the $k-1$ largest edges in the MST, resulting in k connected subtrees (the clusters). If the data set is small enough to apply clusiVAT without the sampling step, this is essentially the single linkage algorithm. The essential step in clusiVAT for $N > 10,000$ is to extend this k-partition of D_n non-iteratively to the unlabeled objects in T using the *nearest (object) prototype rule* (NPR) [20, 21]. Pseudocodes for VAT, iVAT, sVAT and clusiVAT are well documented in the literature and hence are not presented here for brevity.

We use the general term $dist(T_i, T_j)$ to represent the distance between trajectories T_i and T_j, and have not specified which distance measure (Euclidean, nondirectional Euclidean, DTW, or nondirectional DTW) to use. The choice of distance measure depends on the trajectory sampling scheme and is discussed on a case by case basis for the experiments in Section 7.

To illustrate trajectory clustering using iVAT and clusiVAT, Figs. 3 and 4 show two experiments based on building the input distance matrix with nondirectional Euclidean distances as in (2). Fig. 3 shows a small dataset of 400 trajectories represented by $m = 16$ 2-dimensional samples similar to the trajectories T_1 - T_4 described in Section 3. These trajectories are divided into 4 labelled subsets of 100 trajectories each as shown in view 3(a). The VAT image in view (b) indicates the presence of 4 possible clusters by 4 (hard to see) dark blocks along the diagonal. These dark blocks are much clearer (but you have to zoom the image to see the two sub-blocks within the larger dark block) in Fig. 3(c), which is the iVAT image. Finally view (d) shows four



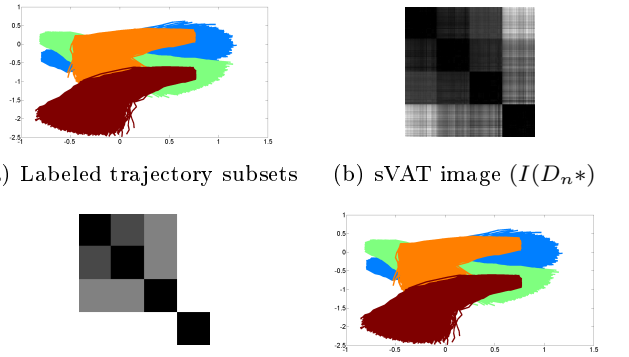
(a) Labeled trajectory subsets (b) VAT image ($I(D*)$)

(c) iVAT image ($I(D'^*)$) (d) SL trajectory clusters

Fig. 3: Labeled Trajectories, VAT, iVAT and clustered trajectories for a small dataset of 4 subsets having 100 trajectories each

clusters found by 3 single linkage cuts in the iVAT MST, which are exactly the same as the labelled trajectories.

Fig. 4 shows the big data version of the clustering experiments, where now we have N =100,000 trajectories split into 4 labeled subsets of 25,000 paths each (Fig. 4(a)). View 4(b) shows the sVAT image of D_n, based on $n = 500$ sample trajectories taken from the N =100,000 big dataset. Fig. 4(c) is the clusiVAT image of the sample D_n, which clearly suggests that the dataset contains 4 clusters of trajectories. Fig. 4(d) shows the trajectories belonging to the four clusters of trajectories obtained by the clusiVAT clustering algorithm.



(a) Labeled trajectory subsets (b) sVAT image ($I(D_n*)$

(c) clusiVAT image ($I(D_n'^*)$ (d) clusiVAT trajectory clusters

Fig. 4: Labeled trajectories, sVAT, clusiVAT and clustered trajectories for N =100,000 labeled trajectories divided in 4 subsets

If the dataset is complex or the clusiVAT clusters are intermixed with each other and contain a number of anomalies, then cutting the $k-1$ largest edges of the MST to obtain k clusters is not always a good strategy as the anomalies, which are (presumably) at a large distance from the normal clusters, might be attached to the $k-1$ largest edges of the MST. A more useful approach in such a scenario is to manually select the dark blocks along the diagonal, find the sample trajectories representing each dark block, and use the nearest prototype rule to find those trajectories in the dataset T whose nearest trajectory belongs to the sample trajectories of the cluster.

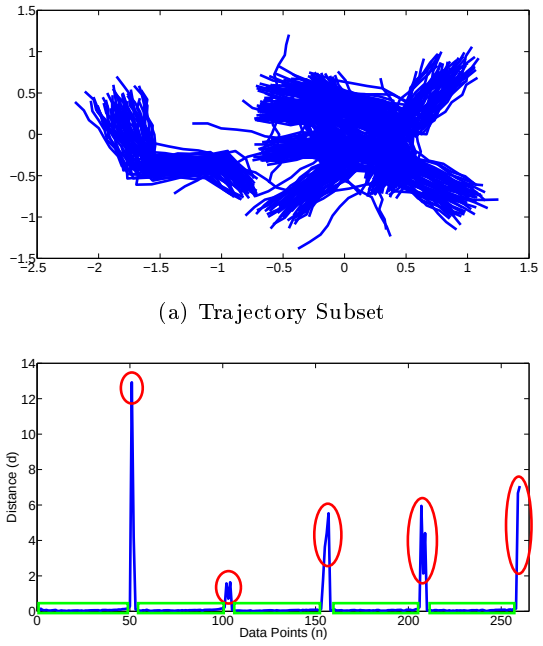
5 iVAT+/clusiVAT+: Anomaly Detection using iVAT/clusiVAT Algorithm

Anomaly detection can be regarded as a special case of data clustering in which the objects in clusters that are too far from the main clusters, or have too few datapoints, are regarded as anomalies. We use this concept for anomaly detection using the iVAT and clusiVAT algorithms, which are denoted as iVAT+ and clusiVAT+ respectively, where the "+" means the base model *plus* the heuristics in (5,6,7) (described below) which are added to the basic model to make it applicable to anomaly detection. Clusters in the "+" models are obtained by cutting the MST using threshold magnitudes ordered by edge distances d in the MST. The cluster boundaries are defined by those indices t, which satisfy

$$d_t > \alpha, \tag{5}$$

where α is a parameter that controls how far two groups of datapoints should be from each other to be considered as separate clusters. Small values of α create tighter boundaries for clusters, while large values of α represent loose cluster boundaries. Next we describe the procedure to find an optimal value of α.

Consider a subset of Test dataset 1 [33] shown in Fig. 5(a), which consists of a total of 260 trajectories divided into 5 clusters of 50 (normal) trajectories plus a set of 10 anomalies. We apply VAT to the dissimi-



(a) Trajectory Subset

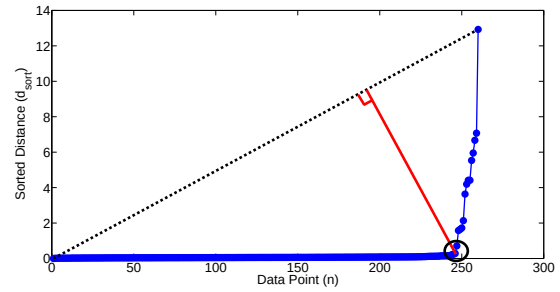
(b) MST cut magnitude (d)

Fig. 5: (a) A subset of Test dataset 1 (containing 260 trajectories) and (b) its VAT generated MST cut magnitude d

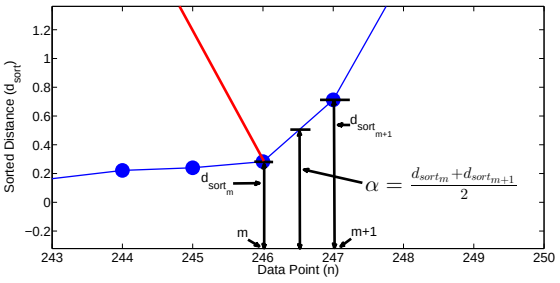
larity matrix for this subset, using the Euclidean distance between each pair of trajectories as a distance

measure. The edge lengths of the VAT generated MST are shown in Fig. 5(b). The distances between trajectories belonging to the same cluster are small, hence edge lengths along the MST consisting of trajectories belonging to the same cluster are also very small (close to zero), as shown by the green rectangles in Fig. 5(b). Edges of the VAT generated MST, that connect two trajectories belonging to different clusters or to anomalies have a much higher edge length than the intra-cluster edges. These inter-cluster edges and edges connecting anomalous trajectories are enclosed by red ellipses in Fig. 5(b). We want a value of α that is larger than the highest intra-cluster edge length, but smaller than the minimum value of inter-cluster edge lengths.

To find the optimal value of α, we sort the edge lengths in the MST in increasing order as shown in Fig. 6(a). The plot of d_{sort} has a high-curvature "elbow"



(a) Sorted d



(b) Zoom elbow

Fig. 6: Finding an optimal value of α using "elbow" detection in the sorted values of edge lengths in the MST

separating the low length intra-cluster edges, characterized by a slow growth in the length, from the inter-cluster edges and edges connecting anomalous trajectories, characterized by a final quick growth of d_{sort}. The elbow can be identified by searching for the point on the plot that has the largest perpendicular distance from a line connecting the two extremes of the plot (shown by the black dotted line in Fig. 6(a)). The encircled elbow part of view (a) is zoomed and shown in Fig. 6(b). Let m be the index of the elbow point. In this case $d_{sort_{1:m}}$ represents the intra-cluster edge lengths of the VAT generated MST, and $d_{sort_{m+1:N}}$ represents the inter-cluster edge length and the lengths of the edges connecting anomalous trajectories, where N is the total number of trajectories. Hence we choose the optimal value of α as the average of the longest intra-cluster edge (d_{sort_m}) and shortest inter-cluster edge ($d_{sort_{m+1}}$), giving the following equation

$$\alpha = \frac{d_{sort_m} + d_{sort_{m+1}}}{2}. \tag{6}$$

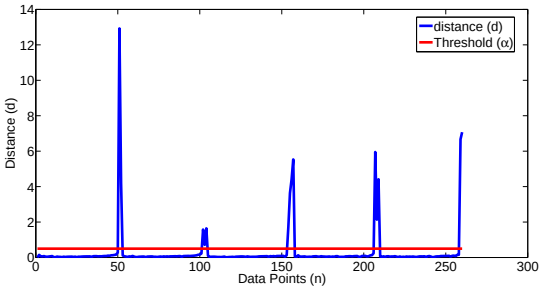
For the dataset shown in Fig. 5(a), the index of the elbow point is $m = 246$, giving the length of the longest intra-cluster edge as $d_{sort_{246}} = 0.2828$, and the shortest of the inter-cluster edges and edges connecting anomalous trajectories to be $d_{sort_{247}} = 0.7124$. Hence we obtain $\alpha = \frac{0.2828+0.7124}{2} = 0.4976$. Using this value of α, we cut those edges of the MST generated by iVAT/ clusiVAT that satisfy (5), resulting in one or more connected subtrees (clusters), which are denoted by ST_i, $i \geq 1$. For clusiVAT, finding the clusters would require an additional step of assigning each non-sampled trajectory to the nearest cluster using the nearest prototype rule. The value of α is data dependent, and is computed automatically for each new input set as part of the implementation of iVAT+ and clusiVAT+. However, we will depart from this strategy in several examples presented below to facilitate direct comparisons with trajectory analysis results from several other papers.

We use the number of trajectories in a cluster as a measure of normality for the cluster. A cluster ST_i is regarded as anomalous if it satisfies the following condition

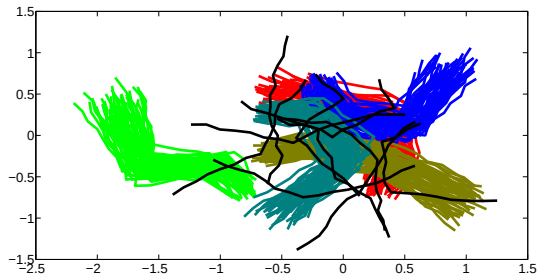
$$|ST_i| < \lfloor \beta \times N \rfloor, \tag{7}$$

where $|ST_i|$ is the number of trajectories in cluster ST_i, $0 \leq \beta \leq 1$ is a user defined parameter that controls the size of a trajectory cluster that is to be declared anomalous and N is the total number of trajectories in the dataset. Large values of β (close to 1) cause even large groups of datapoints that are far from other clusters to be declared anomalous. As β decreases, anomalous clusters eventually become part of the normal cluster structure, and only isolated trajectories or a group of a few trajectories remain as anomalous. The choice of β is dependent on the application as well. For applications which recognize a (relatively) small proportion of similar trajectories as a separate cluster, β can be set to a smaller value (say 0.01 – 0.03), whereas for applications where the expected number of anomalous trajectories is large, β needs to be higher (of the range 0.05 – 0.1).

Fig. 7(a) shows the MST cut magnitude d and the cut threshold α for the 260 trajectory data set shown in Fig. 5(a). All the edges of the MST having lengths



(a) MST cut magnitude d and edge cut threshold α



(b) 5 trajectory clusters and 10 anomalous trajectories (in black)

Fig. 7: Cutting the edges of VAT generated MST to obtain clustering based labels

greater than $\alpha = 0.4976$ are cut to generate 15 subtrees, 5 of which have 50 trajectories each and the remaining 10 have one trajectory each. We choose β so that any cluster having less than (say) 5% of the total number of trajectories (N) is declared anomalous, so that in this case $\beta = 0.05$. Any trajectory cluster ST_i satisfying $|ST_i| < \beta \times N = 0.05 \times 260 = 13$ is declared anomalous. Hence the 10 singleton trajectories are declared as anomalies. Fig. 7(b) shows the final results of our iVAT+ algorithm, where trajectories belonging to 5 different labeled subsets are shown in different colors and the 10 anomalous trajectories are shown in black.

Fig. 8 is an architectural diagram for clustering and anomaly detection using iVAT and clusiVAT. The first test is to determine if a dataset is small or big based on the number of trajectories in it. The threshold for this selection is chosen to be $N = 10,000$. For small datasets, iVAT provides a visual image that enables us to estimate the number of clusters in the data, and then its MST is cut to obtain single linkage clusters. When the operations at equations (5) and (7) are added, iVAT becomes iVAT+ for anomaly detection. For large datasets, clusiVAT and clusiVAT+ are used for clustering and anomaly detection in the same way.

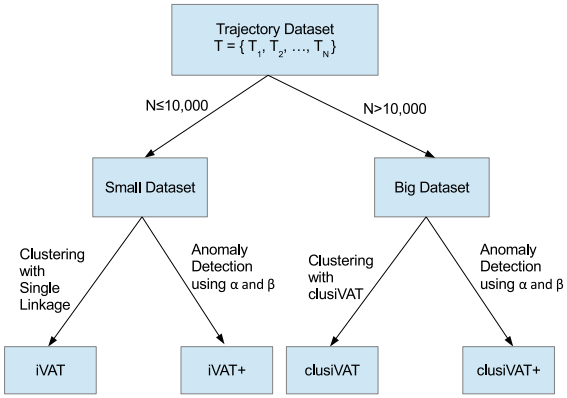


Fig. 8: Architecture diagram for clustering and anomaly detection using iVAT and clusiVAT

6 Time Complexity

In this section we discuss the time complexity of iVAT+ and clusiVAT+. These algorithms apply iVAT/clusiVAT to trajectory pairs using DTW distance. For trajectories of length l and m, the standard DTW algorithm is $O(l \times m)$, but there are approximate DTW algorithms like fast DTW [40] with time complexity that is linear in the average length of the trajectories, viz., $O(l+m)$. Let data set T contain N trajectories. There are two cases:
(i) $N \leq 10,000$. We use iVAT, time complexity $O(N^2)$, which includes the time needed to compute the $N \times N$ DTW distance matrix D_N.
(ii) $N > 10,000$. We use clusiVAT. Step 1 requires time $O(k')$ to select k' distinguished trajectories from k' (nearly) equal-sized subsets of T. Then a random sample of size n which approximately retains the structure in T is drawn from the k' subsets. Then iVAT is applied to D_n, with time complexity $O(n^2)$, which includes the time needed to construct the DTW distance matrix D_n.

The essential step in iVAT+ or clusiVAT+ is sorting the MST cut magnitude (d) before thresholding as described in Section 5. The sorting can be done efficiently by Quicksort [15], which has an average complexity of $O(n \times log(n))$. Summarizing, the overall complexity of iVAT+ and clusiVAT+ is, assuming l and m are quite small compared to N and n, $O(N^2)$ in case (i); and $O(n^2)$ in case (ii).

7 Numerical Experiments

We have performed three sets of experiments. The first two experiments are anomalous trajectory detection on publicly available synthetic trajectory datasets [22, 23, 33]. The third experiment is trajectory clustering and anomaly detection using the real life MIT trajectories dataset [42] consisting of 40,453 trajectories of vehicles and pedestrians from a parking lot scene collected over a period of 5 days.

7.1 Outlier detection experiment for synthetic trajectory datasets

This experiment is performed on 2 datasets consisting of synthetic trajectories (two-dimensional trajectories of length $m = 16$ each), first published by Piciarelli et al. [33] and later reproduced by Laxhammar and Falkman [22]. These two datasets are publicly available at [1]. Dataset 1 consists of trajectory subsets having a fixed number of clusters and fixed number of anomalies, whereas Dataset 2 has a variable number of clusters and a variable number of anomalies (both ranging 1 to 10). Sampling information is not available, so we consider them to be sampled at equal time intervals.

7.1.1 Fixed numbers of clusters and anomalies

Test dataset 1 consists of 1000 randomly generated undirected trajectory subsets, each of which contains $n = 260$ two-dimensional trajectories of length $m = 16$. Hence each trajectory T_i is represented by 16 two dimensional points, $T_i = (t_i^1, t_i^2, ..., t_i^{16})$, $t_i^j \in \mathbb{R}^2, 1 \leq i \leq 260, 1 \leq j \leq 16$. Of the 260 trajectories, the first 250 belong to five different subsets and are labelled as normal. The remaining 10 are stray trajectories that do not belong to any labeled subset and are considered as abnormal. A plot of one of the subsets of this dataset is shown in Fig. 7(b), where trajectories belonging to 5 different labeled subsets are shown in different colors and the 10 anomalous trajectories are shown in black.

In this experiment we use the Euclidean distance between each pair of trajectories as a distance measure to generate a 260×260 distance matrix D for each of the 1000 subsets. Since the trajectories are sampled at equal time intervals, Euclidean distance is reliable enough, so we do no need to use the computationally expensive DTW distance. Each input matrix D is small enough that we can use iVAT on each distance matrix to obtain a reordered distance matrix D'^*, reordering indices P and ordered MST edge magnitudes d. The anomalous trajectories are extracted using the iVAT+ algorithm described in Section 5. The value of α is calculated for each subset using the "elbow detection" procedure described in Section 5. Since the maximum number of anomalous trajectories in each of the 1000 instances of this data is 10 and the total number of trajectories in each subset is $N = 260$, we set the parameter $\beta = 0.04$ so that a cluster ST_i with $|ST_i| < \lfloor \beta \times N \rfloor = \lfloor 0.04 \times 260 \rfloor = \lfloor 10.4 \rfloor = 10$ would be considered anomalous.

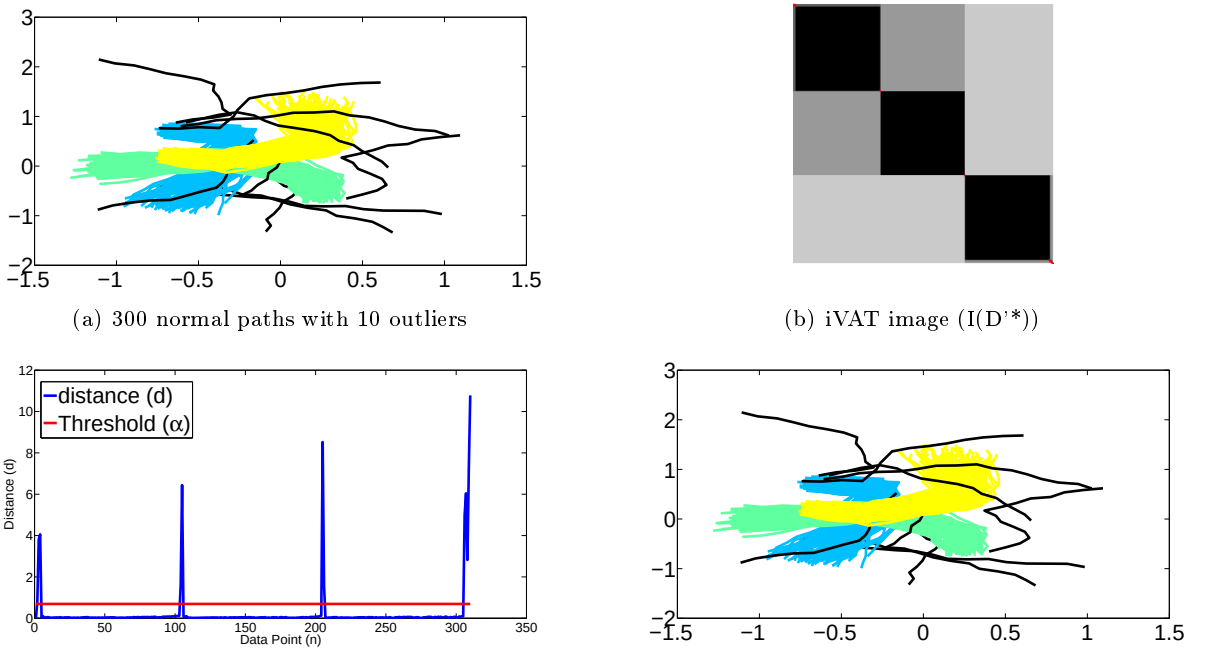
The average value of true positive (anomalies) and true negative (non-anomalies) for the 1000 subsets of test dataset 1 is 9.72 (out of 10) and 249.09 (out of 250) respectively. The partition accuracy (PA) of anomaly vs non-anomaly is

$$\begin{aligned} \text{PA} &= \frac{\text{true positive + true negative}}{\text{total no. of trajectories}} \times 100 \\ &= \frac{9.72 + 249.09}{260} \times 100 \\ &= 99.54\% \end{aligned} \tag{8}$$

Next, we compare the performance of our iVAT+ anomaly detection scheme to some previously reported results. For example, [33] uses a one-class SVM to identify anomalous trajectories, whereas [22] uses a *similarity based nearest neighbour conformal anomaly detector* (SNN-CAD) which is a parameter-light algorithm for on-line learning and anomaly detection with a well calibrated false alarm rate. In order to compare our method with those proposed in [33] and [22], we perform the following experiments in a slightly different manner. These two papers express their results in the form of error scores for each method by averaging over the number of normal trajectories among the 10 trajectories that are declared as outliers. The accuracy is then calculated by subtracting the error rate from 100%. Hence to compare with these two papers, we do not need to find the optimal value of α, but we need to find that value of α for which the number of anomalies is 10. To get the results in this format, we set the threshold $\beta = 0.04$ and α to the maximum value of the MST cut magnitude d and keep decreasing it until the number of declared anomalous trajectories becomes 10. The results in Table 3 are for this experiment setup. The mean value (over the 1000 sets in this dataset) of the error scores is 2.38%, giving an accuracy of $100 - 2.38 = 97.62\%$, which is better than the accuracy obtained by most of the methods reported in Table I in [22], which provides the average accuracy results obtained by the various outlier measures presented in [18, 22, 33]. These results are reproduced here as part of our Table 3 for comparison.

Table 3: Average accuracy on dataset 1 for different outlier measures presented in Table I in [22]

	No. of most similar neighbours considered				
Outlier measure	k = 1	k = 2	k = 3	k = 4	k = 5
SNN with $S = \boldsymbol{\delta_h}$ [22]	96.59%	97.12%	97.05%	96.84%	96.72%
SNN with $S = \delta_h$ [22]	97.28%	97.66%	97.63%	97.57%	97.37%
SVM [33]	96.30%				
Discords [18]	97.04%				
iVAT+	**97.62%**				



(a) 300 normal paths with 10 outliers

(b) iVAT image (I(D'*))

(c) MST cut magnitude order

(d) Detected anomalies (black) and clusters (different colors)

Fig. 9: Anomalous trajectory detection example

7.1.2 Variable number of clusters and anomalies

Test dataset 2 consists of 100 different experimental cases, each with a different number of groups of 100 normal trajectories and outliers (both ranging from 1 to 10). For each case, ten different training/test sets were created, for a total of 2000 datasets. Since the iVAT algorithm does not require any training, we consider them as 2000 separate test subsets. iVAT+ is used again for anomaly detection because the maximum value of $N = 10(\text{clusters}) \times 100(\text{trajectories per cluster}) + 10(\text{anomalies}) = 1010 < 10,000$. Euclidean distance is used again as the trajectories are sampled at equal time intervals.

Fig. 9 shows an example of anomalous trajectory detection using iVAT+ for this type of data. View (a) shows an example dataset consisting of 3 labeled subsets having 100 trajectories each and 10 outliers giving a total of 310 trajectories. The three dark diagonal blocks in the iVAT image ($I(D'^*)$) in view 9(b) suggest that there are three clusters in D. At 1600x magnification, you can see the 10 anomalies in this image, visible as small dark blocks of 1-2 pixels (marked as red) at both ends of each of the three big dark blocks. The highest ratio of the number of anomalies to the total number of trajectories in a subset is obtained for the subset having 10 anomalies and 1 subset having 100 normal trajectories giving a maximum limit of $\beta = \frac{10}{1\times100+10} = 0.091$. Similarly the minimum limit of β is obtained for the subset having 1 anomaly and 10 subsets having 100 normal trajectories, giving $\beta = \frac{1}{10\times100+1} = 0.001$. Hence for the iVAT+ experiment we choose the value of β to be midway between the maximum and minimum limits, giving $\beta = \frac{0.091+0.001}{2} = 0.046$. The value of α is found using the elbow detection procedure described in Section 5 for each subset. Fig. 9(c) shows the plot of the iVAT ordered MST cut magnitude (d) and a red horizontal line, $y = \alpha$ separating the normal clusters from

Table 4: iVAT+: False alarms (%) for test dataset 2

Anomalies	No. of clusters									
	1	2	3	4	5	6	7	8	9	10
1	1.2	0.2	0.3	0.5	0.4	0.6	0.4	0.4	0.5	0.6
2	0	0	0.2	0.3	0.4	0.4	0.2	0.3	0.5	0.3
3	0	0.2	0.2	0.3	0.3	0.3	0.5	0.3	0.3	0.6
4	0	0.1	0.3	0.3	0.3	0.5	0.6	0.3	0.5	0.4
5	0	0.4	0.4	0.5	0.3	0.5	0.5	0.5	0.5	0.6
6	0.2	0.2	0.2	0.2	0.5	0.4	0.3	0.3	0.4	0.5
7	0	0.2	0.1	0.3	0.3	0.4	0.3	0.4	0.3	0.4
8	0	0.4	0.2	0.3	0.3	0.5	0.3	0.5	0.4	0.5
9	0.1	0.2	0.2	0.2	0.3	0.5	0.5	0.7	0.2	0.6
10	0.1	0	0.1	0.4	0.2	0.4	0.5	0.4	0.4	0.6

Table 5: iVAT+: True positives (%) for test dataset 2

Anomalies	No. of clusters									
	1	2	3	4	5	6	7	8	9	10
1	100	100	100	100	90	100	100	100	100	100
2	100	100	100	90	100	100	95	100	100	100
3	97	100	100	100	97	100	100	100	100	97
4	100	98	95	98	95	95	100	92	100	95
5	100	98	98	100	98	94	100	100	90	98
6	100	100	97	100	97	95	97	93	98	93
7	100	100	97	100	93	97	97	96	94	94
8	99	99	99	99	100	100	99	99	99	99
9	100	98	98	98	94	98	98	100	97	99
10	100	100	98	99	97	99	96	98	100	98

anomalies. The trajectories belonging to the 3 different clusters are shown by 3 different colors and the anomalies are shown by black curves in Fig. 9(d).

Tables 4 and 5 show the average percentage values of false alarms and true positives for the 1000 test subsets of dataset 2. Comparing these results with those given in Tables III and IV in [33], iVAT+ based anomaly detection presents fewer false alarms and correctly classifies more anomalies as compared to the method given in [33] in almost all cases. The average true positive detection rate over all cases in Table 5 is 98%, which is higher than the 92% reported in [33].

7.2 MIT trajectory dataset experiment

The MIT trajectories dataset [42] is a standard, publicly available dataset consisting of 40,453 trajectories of vehicles and pedestrians from a parking lot scene collected over a period of 5 days. Fig. 10 shows the parking lot scene used for this experiment. We exclude from the $40,453$ trajectories those trajectories that have any point belonging to the blue shaded area, as this represents the buildings and does not include pedestrian or car trajectories. This leaves us with $N = 26,039$ trajectories belonging to the non-shaded area in Fig. 10 whose lengths range from 100 to 1100 sample points per trajectory. We resample these variable length trajectories to obtain 20 equidistant 2-dimensional points, hence each trajectory is represented by the x and y coordinates of 20 2-dimensional points.

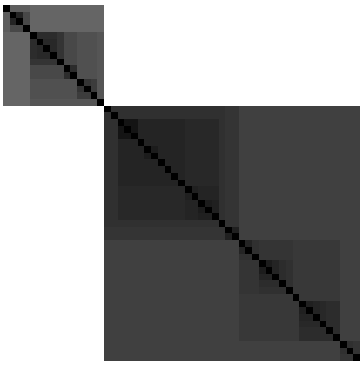


Fig. 10: Camera view for the MIT parking lot trajectory data

Since $N > 10,000$, we use clusiVAT for clustering and clusiVAT+ for anomaly detection. To cope with the different directions of trajectories, we use a novel two-stage form of clusiVAT clustering. In the first stage,



(a) Cluster sample trajectories (Stage 1)



(b) clusiVAT image (stage 2)



(c) Subcluster 1 (stage 2)



(d) Subcluster 2 (stage 2)

Fig. 12: A sample cluster and subclusters obtained using the two-stage clusiVAT clustering

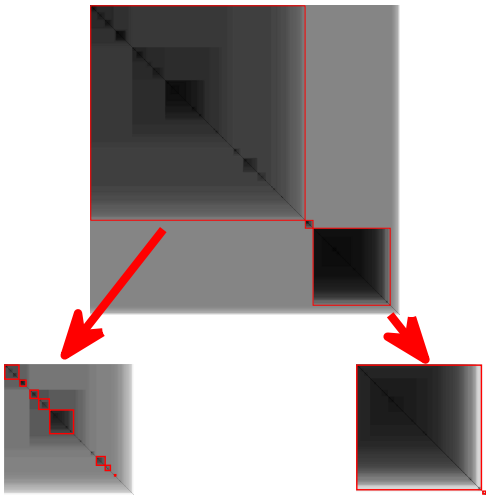
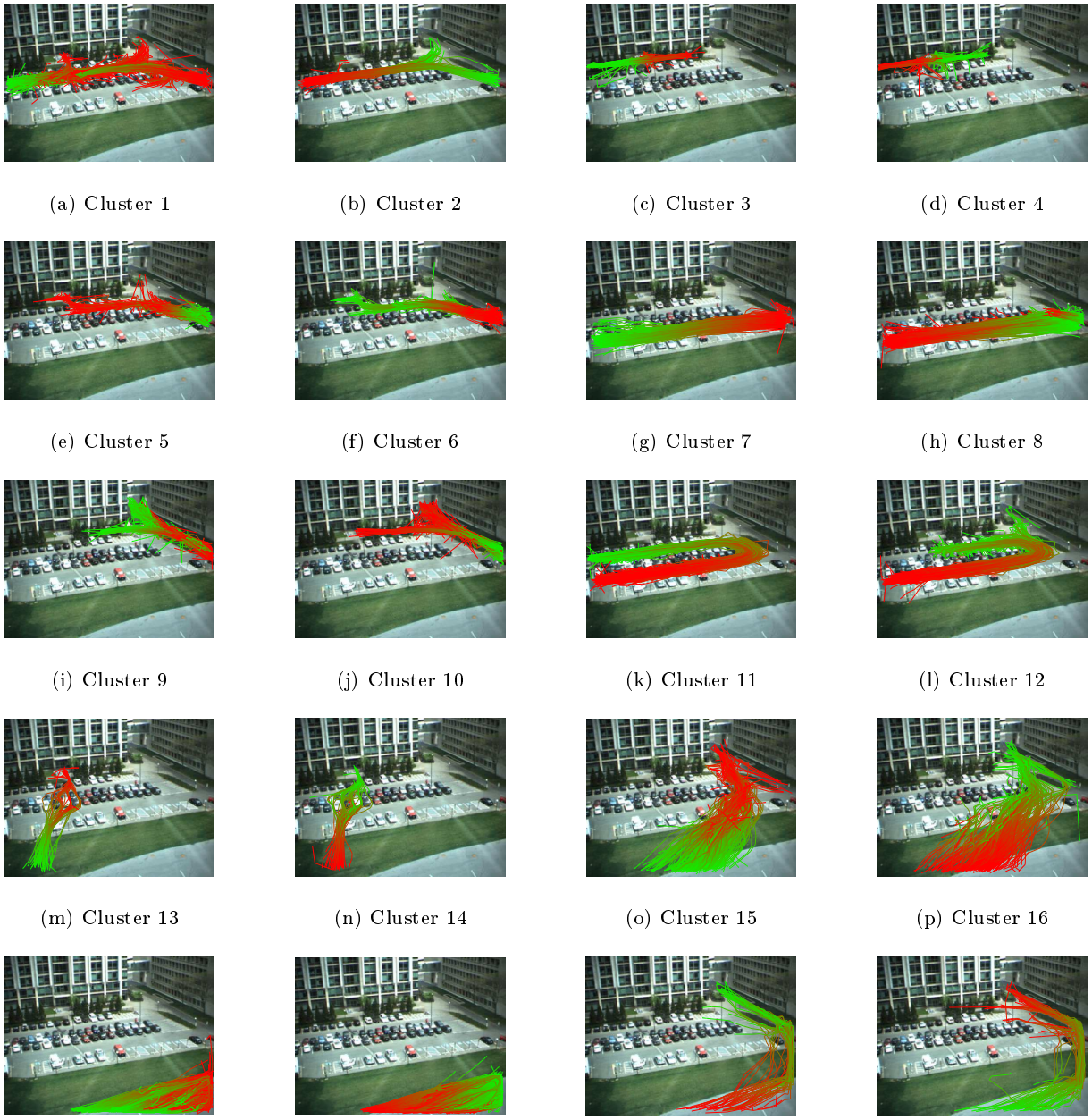


Fig. 11: Reordered distance matrix (top) and zoomed first and third dark blocks (bottom) using clusiVAT algorithm

nondirectional DTW is used as a distance measure to cluster the 26,039 trajectories, which ensures that trajectories that have opposite starting and finishing points, but follow similar paths, are clustered in the same group. The top image in Fig. 11 shows the image of the clusiVAT reordered distance matrix ($D_n^{'*}$) for $n = 1,500$ and $k' = 30$ (see Inputs of the siVAT Algorithm for k' in [13, 20, 21]). This image seems to have three primary dark blocks, and embedded in them, a fine structure that has many more. The zoomed images of the first and third dark blocks shown in the lower part of Fig. 11 reveal the presence of 8 and 2 dark sub-blocks representing imbedded subclusters. These dark blocks are marked by red rectangles for clarity.

In the next stage, we use DTW as the distance measure for each cluster of trajectories obtained in the first step in order to separate the trajectories going in opposite directions using the clusiVAT algorithm. Fig. 12 shows an example of the clusters obtained using two-stage clusiVAT clustering. View (a) in Fig. 12 shows



(a) Cluster 1 (b) Cluster 2 (c) Cluster 3 (d) Cluster 4

(e) Cluster 5 (f) Cluster 6 (g) Cluster 7 (h) Cluster 8

(i) Cluster 9 (j) Cluster 10 (k) Cluster 11 (l) Cluster 12

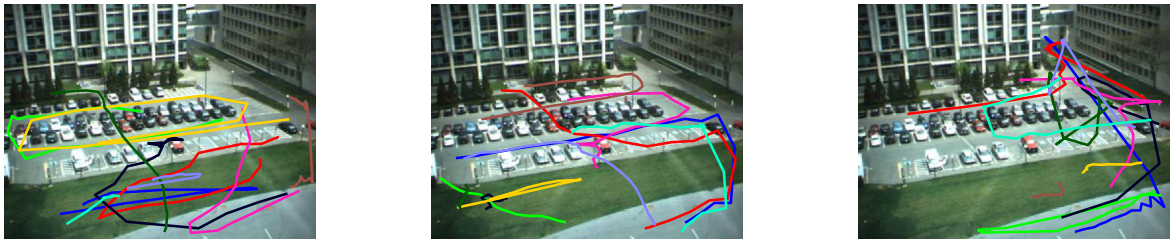
(m) Cluster 13 (n) Cluster 14 (o) Cluster 15 (p) Cluster 16

(q) Cluster 17 (r) Cluster 18 (s) Cluster 19 (t) Cluster 20

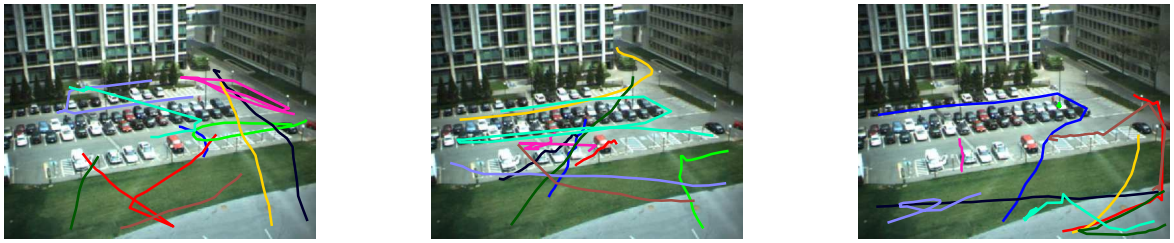
Fig. 13: Twenty MIT trajectory dataset clusters obtained using the two-stage clusiVAT algorithm

the sample trajectories belonging to a cluster obtained in the first stage, which represents the tracks following a particular path in each direction. The cluster thus obtained is further clustered in the second stage using the DTW distance to give the clusiVAT reordered image shown in Fig. 12(b), which gives the two subclusters of the original cluster shown in views (c-d) of Fig. 12. The trajectories belonging to these two subclusters have similar paths but opposite directions as shown by the opposite locations of the green (starting) and red (ending) parts of the trajectories.

The 20 trajectory clusters obtained using our two-stage clusiVAT process are shown in Fig. 13. Since the dataset is not labelled, we cannot numerically compare our results with results previously published in literature [7,43], however, these clusters are more natural and informative than the ones previously reported as explained below. The trajectory clusters in Fig. 13 show



(a) 1-10 (b) 11-20 (c) 21-30



(d) 31-40 (e) 41-50 (f) 51-60

Fig. 14: Top 60 anomalous trajectories using clusiVAT+ algorithm

the most common path taken by pedestrians and vehicles along the road and footpath. All the trajectory clusters obtained from the dataset are not shown in [7] and our clusiVAT algorithm is able to find those shown in [7]. Some of the clusters reported in [43] are repeated and some do not represent any particular path taken by pedestrians. For example, clusters 14 and 15 are a subpart of cluster 13 in Fig. 10 of [43], so they should not be represented as separate clusters; our method on the other hand divides the trajectories lying in this region into two clusters based on directionality, i.e., clusters 17 and 18 in Fig. 13. Clusters 18-21 in Fig. 10 of [43] do not specify any particular path taken by pedestrians but comprise trajectories passing through the center of the camera view, which is not a good rationale for clustering trajectories. Our approach also discovers two new clusters, cluster 19 and 20 in Fig. 13, which were not reported in [43].

To find anomalous trajectories in the MIT trajectory dataset we use clusiVAT+ with nondirectional DTW as the distance measure. Since $N = 26,039$ is relatively large, we use $\beta = 0.001$, so that any trajectory cluster ST_i with $|ST_i| < \lfloor \beta \times N \rfloor = \lfloor 0.001 \times 26039 \rfloor = 26$ trajectories is declared anomalous. To find the top anomalous trajectories, we set α to be the maximum value of the MST cut magnitude d and decrease it until we get the top 60 anomalies, which are shown in Fig. 14. These represent a group of trajectories far away from the remaining clusters that follow a zig-zag or very long path.

8 Conclusions

In this paper we presented a clustering and anomaly detection framework for digitized video trajectories. We develop iVAT+ for anomaly detection in the small data case ($N \leq 10,000$), and applied it to two publicly available synthetic trajectory datasets [22,23,33]. The mean error rate is 2.38%, giving an accuracy of 97.62%, which is better than the accuracy previously reported in the literature on dataset 1. For the second dataset the average true positive detection rate is 98%, which is higher than the 92% reported earlier. We also perform clustering and anomaly detection on a real life MIT trajectories dataset of vehicles and pedestrians from a parking lot scene [42]. We performed a two-stage clusiVAT experiment, which uses nondirectional DTW as a distance measure in the first stage to cluster all the trajectories disregarding their orientation. In the second stage the clusters obtained from the first stage are further clustered using DTW based clusiVAT to separate the trajectories heading in the opposite direction. The trajectory clusters obtained using our two-stage clusiVAT process are more natural and informative than those that have previously been reported. We also detected anomalous trajectories in this dataset using the clusiVAT+ algorithm.

The trajectory clusters and anomalies obtained using our clusiVAT/clusiVAT+ framework can help in the high level interpretation of a scene (crowd behaviour modelling), and also as feedback for a low-level (individual) tracking system. Anomalous trajectory detection can be used as an alarm system for a human supervisor. The clusters obtained using the clusiVAT algorithm can be used to generate a probabilistic model of typical trajectory behavior to focus a surveillance system's attention to the events of interest, such as activity classification, abnormality detection and activity prediction.

9 Acknowledgement

We acknowledge the support from the Australian Research Council (ARC) Linkage Project grant (LP120100529), the ARC Linkage Infrastructure, Equipment and Facilities scheme (LIEF) grant (LF120100129), the EU-FP7 SOCIOTAL grant and National ICT Australia (NICTA).

References

1. Synthetic trajectory dataset. `http://avires.dimi.uniud.it/papers/trclust/dataset2.zip`. Accessed: 2014-12-23.
2. I. Ali and M. N. Dailey. Multiple human tracking in high-density crowds. *Image and Vision Computing*, 30(12):966 – 977, 2012.
3. O. Arandjelović. Contextually learnt detection of unusual motion-based behaviour in crowded public spaces. In *Computer and Information Sciences II*, pages 403–410. Springer London, 2012.
4. Y. Benezeth, P. Jodoin, V. Saligrama, and C. Rosenberger. Abnormal events detection based on spatio-temporal co-occurences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2458–2465, June 2009.
5. J. Bezdek and R. Hathaway. VAT: A tool for visual assessment of (cluster) tendency. *International Joint Conference on Neural Networks (IJCNN)*, pages 2225–2230, 2002.
6. F. Bousetouane, L. Dib, and H. Snoussi. Improved mean shift integrating texture and color features for robust real time object tracking. *The Visual Computer*, 29(3):155–170, 2013.
7. L. Brun, A. Saggese, and M. Vento. Learning and classification of car trajectories in road video by string kernels. In *International Conference on Computer Vision Theory and Applications*, volume 1, pages 709–714, 2013.
8. A. Chan, V. Mahadevan, and N. Vasconcelos. Generalized stauffer–grimson background subtraction for dynamic scenes. *Machine Vision and Applications*, 22(5):751–766, 2011.
9. Y. Cong, J. Yuan, and J. Liu. Sparse reconstruction cost for abnormal event detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3449–3456, June 2011.
10. Y. Cong, J. Yuan, and Y. Tang. Video anomaly search in crowded scenes via spatio-temporal motion context. *IEEE Transactions on Information Forensics and Security*, 8(10):1590–1599, Oct 2013.
11. X. Cui, Q. Liu, M. Gao, and D. Metaxas. Abnormal detection using interaction energy potentials. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3161–3167, June 2011.
12. Z. Fu, W. Hu, and T. Tan. Similarity based vehicle trajectory clustering and anomaly detection. In *IEEE International Conference on Image Processing (ICIP)*, volume 2, pages II–602–5, Sept 2005.
13. R. Hathaway, J. Bezdek, and J. Huband. Scalable visual assessment of cluster tendency for large data sets. *Pattern Recognition*, 39:1315–1324, 2006.
14. T. Havens and J. Bezdek. An efficient formulation of the improved visual assessment of cluster tendency (iVAT) algorithm. *IEEE Transactions on Knowledge and Data Engineering*, 24(5):813–822, 2012.
15. C. A. R. Hoare. Algorithm 64: Quicksort. *Commun. ACM*, 4(7):321, July 1961.
16. F. Jiang, Y. Wu, and A. Katsaggelos. A dynamic hierarchical clustering method for trajectory-based unusual video event detection. *IEEE Transactions on Image Processing*, 18(4):907–913, April 2009.
17. F. Jiang, J. Yuan, S. A. Tsaftaris, and A. K. Katsaggelos. Anomalous video event detection using spatiotemporal context. *Computer Vision and Image Understanding*, 115(3):323 – 333, 2011.
18. E. Keogh, J. Lin, and A. Fu. Hot sax: Efficiently finding the most unusual time series subsequence. In *IEEE International Conference on Data Mining*, ICDM '05, pages 226–233, 2005.
19. J. Kim and K. Grauman. Observe locally, infer globally: A space-time mrf for detecting abnormal activities with incremental updates. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2928, June 2009.
20. D. Kumar, J. Bezdek, M. Palaniswami, S. Rajasegarar, C. Leckie, and T. Havens. A hybrid approach to clustering in big data. *IEEE Transactions on Cybernetics*, PP(99):1–1, 2015.
21. D. Kumar, M. Palaniswami, S. Rajasegarar, C. Leckie, J. Bezdek, and T. Havens. clusiVAT: A mixed visual/numerical clustering algorithm for big data. In *IEEE International Conference on Big Data*, pages 112–117, Oct 2013.
22. R. Laxhammar and G. Falkman. Sequential conformal anomaly detection in trajectories based on hausdorff distance. In *International Conference on Information Fusion (FUSION)*, pages 1–8, July 2011.
23. R. Laxhammar and G. Falkman. Online learning and sequential anomaly detection in trajectories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6):1158–1173, June 2014.
24. W. Li, V. Mahadevan, and N. Vasconcelos. Anomaly detection and localization in crowded scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):18–32, Jan 2014.
25. X. Li, J. Han, S. Kim, and H. Gonzalez. ROAM: Rule and Motif-Based Anomaly Detection in Massive Moving Object Data Sets. In *SIAM International Conference on Data Mining*, 2007.
26. R. Martin and O. Arandjelović. Multiple-object tracking in cluttered and crowded public spaces. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, R. Chung, R. Hammound, M. Hussain, T. Kar-Han, R. Crawfis, D. Thalmann, D. Kao, and L. Avila, editors, *Advances in Visual*

Computing, volume 6455 of *Lecture Notes in Computer Science*, pages 89–98. Springer Berlin Heidelberg, 2010.
27. R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 935–942, June 2009.
28. B. Morris and M. Trivedi. A survey of vision-based trajectory learning and analysis for surveillance. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(8):1114–1127, Aug 2008.
29. B. Morris and M. Trivedi. Learning trajectory patterns by clustering: Experimental studies and comparative evaluation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 312–319, June 2009.
30. A. Naftel and S. Khalid. Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space. *Multimedia Systems*, 12(3):227–238, 2006.
31. D. S. Pham, O. Arandjelovic, and S. Venkatesh. Detection of dynamic background due to swaying movements from motion features. *IEEE Transactions on Image Processing*, 24(1):332–344, Jan 2015.
32. C. Piciarelli and G. Foresti. Anomalous trajectory detection using support vector machines. In *IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 153–158, Sept 2007.
33. C. Piciarelli, C. Micheloni, and G. Foresti. Trajectory-based anomalous event detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1544–1554, Nov 2008.
34. L. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.
35. Y. Rao. Automatic vehicle recognition in multiple cameras for video surveillance. *The Visual Computer*, 31(3):271–280, 2015.
36. M. Rodriguez, J. Sivic, I. Laptev, and J. Y. Audibert. Density-aware person detection and tracking in crowds. In *International Conference on Computer Vision (ICCV)*, 2011.
37. M. Roshtkhari and M. Levine. Online dominant and anomalous behavior detection in videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2611–2618, June 2013.
38. I. Saleemi, K. Shafique, and M. Shah. Probabilistic modeling of scene dynamics for applications in visual surveillance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(8):1472–1485, Aug 2009.
39. V. Saligrama, J. Konrad, and P. Jodoin. Video anomaly identification. *IEEE Signal Processing Magazine*, 27(5):18–33, Sept 2010.
40. S. Salvador and P. Chan. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, 11(5):561–580, Oct. 2007.
41. S. Vishwakarma and A. Agrawal. A survey on activity recognition and behavior understanding in video surveillance. *The Visual Computer*, 29(10):983–1009, 2013.
42. X. Wang, K. Ma, G. W. Ng, and W. Grimson. Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models. *International Journal of Computer Vision*, 95(3):287–312, 2011.
43. X. Wang, K. T. Ma, G. W. Ng, and W. Grimson. Trajectory analysis and semantic region modeling using a nonparametric bayesian model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008.
44. B. Zhao, L. F. Fei, and E. P. Xing. Online detection of unusual events in videos via dynamic sparse coding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3313–3320. IEEE Computer Society, 2011.
45. Y. Zhou, S. Yan, and T. Huang. Detecting anomaly in videos from trajectory similarity analysis. In *IEEE International Conference on Multimedia and Expo*, pages 1087–1090, July 2007.



Dheeraj Kumar is currently a Ph.D. student at the Department of Electrical and Electronic Engineering, The University of Melbourne. He received his B.Tech-M.Tech Dual degree in Electrical Engineering from IIT Kanpur, India in 2010. His research interests include Big data clustering, Incremental clustering, spatio-temporal estimations, Internet of things, machine learning, pattern recognition, and signal processing.



James Bezdek received the PhD in Applied Mathematics from Cornell University in 1973. Jim is past president of NAFIPS (North American Fuzzy Information Processing Society), IFSA (International Fuzzy Systems Association) and the IEEE CIS (Computational Intelligence Society): founding editor the Int'l. Jo. Approximate Reasoning and the IEEE Transactions on Fuzzy Systems: Life fellow of the IEEE and IFSA; and a recipient of the IEEE 3rd Millennium, IEEE CIS Fuzzy Systems Pioneer, IEEE technical field award Rosenblatt, and the Kampe de Feriet medals. Jim's interests: optimization, pattern recognition, clustering in very large data, co-clustering, visual clustering and cluster validity.



Sutharshan Rajasegarar received his B.Sc. Engineering degree in Electronic and Telecommunication Engineering (with first class honours) in 2002, from the University of Moratuwa, Sri Lanka, and his Ph.D. in 2009 from the University of Melbourne, Australia. He is currently a Research Fellow with the Department of Electrical and Electronic Engineering, The University of Melbourne, Australia. His research interests include wireless sensor networks, anomaly/outlier detection, Spatio-Temporal estimations, Internet of Things (IoT), machine learning, pattern recognition, signal processing and wireless communication.



Christopher Leckie received the B.Sc. degree in 1985, the B.E. degree in electrical and computer systems engineering (with first class honours) in 1987, and the Ph.D. degree in computer Science in 1992, all from Monash University, Australia. He joined Telstra Research Laboratories in 1988, where he conducted research and development into artificial intelligence techniques for various telecommunication applications. In 2000, he joined the University of Melbourne, Australia, where he is currently a professor in the Department of Computing and Information Systems. His research interests include scalable data mining, network intrusion detection, bioinformatics and wireless sensor networks.



Marimuthu Palaniswami received his M.E. from the Indian Institute of Science, India, M.Eng.Sc. from the University of Melbourne and Ph.D. from the University of Newcastle, Australia. He is currently a professor and a Fellow of IEEE. He has published over 400 refereed research papers, and leads one of the largest funded ARC Research Network on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP) programme. He has been grants panel member for NSF, advisory board member for European FP6 grant centre, steering committee member for NCRIS GBROOS and SEMAT, and board member for IT and SCADA companies. He has been funded by several ARC and industry grants (over 40m) to conduct research in sensor network, Internet of Things (IoT), health, environmental, machine learning and control areas. He is representing Australia as a core partner in EU FP7 projects such as SENSEI, SmartSantander, Internet of Things Initiative (IoT-i) and SocIoTal. His research interests include SVMs, Sensors and Sensor Networks, Internet of Things (IoT), Machine Learning, Neural Network, Pattern Recognition, Signal Processing and Control.