Trajectory Anomaly Detection based on Similarity Analysis

1st GF. Quispe-Torres
Universidad Nacional de San Antonio Abad del Cusco
Cusco, Perú
090227@unsaac.edu.pe

2nd Harley Vera-Olivera
Universidade de Brasília
Brasilia, Brazil
harleyve@gmail.com

3rd Lauro Enciso-Rodas
Universidad Nacional de San Antonio Abad del Cusco
Cusco, Perú
lauro.enciso@unsaac.edu.pe

4th Germain Garcia-Zanabria
Universidad Católica San Pablo
Arequipa, Perú
germain.garcia@ucsp.edu.pe

***Abstract*—Automatic trajectory processing has multiple applications, mainly due to the wide availability of the data. Trajectory data have a significant practical value, making possible the modeling of various problems such as surveillance and tracking devices, detect anomaly trajectories, identifying illegal and adverse activity. In this study, we show a comparative analysis of the performance of two descriptors to detect anomaly trajectories. We define Wavelet and Fourier transforms as trajectory descriptors to generate characteristics and subsequently detect anomalies. The experiments emphasize performance in the description in the coefficient feature space. For that, we used unsupervised learning, specifically clustering techniques, to generate subsets and identify which are irregular. The implications of the study demonstrate that it is possible to use descriptors in trajectories for automatic anomaly detection and the use of unsupervised learning methods that automatically segment the required information. The performance and comparative analysis of our study are demonstrated through experiments and a case study considering synthetic and real data sets that leave evidence of our contribution.**

***Index Terms*—Trajectory anomaly detection, trajectory shape descriptor, feature extraction, trajectory clustering**

I. INTRODUCTION

Understanding trajectory dynamics is a challenging problem due to the wide data and Spatio-temporal information. Trajectories analysis play an important role in different areas such as animal tracking [1], air-streams [2], weather prediction [3], traffic flow [4], activity flow [5], sports [6], flight planning [7], and many others. In this field, anomaly behavior may indicate important objects and events in a wide variety of domains [8]. However, this analysis is not a trivial problem due to the sequential analysis, complexity morphology, and parameter calibration of algorithms.

Anomaly trajectory detection is an important problem because it allows identifying trajectories that may indicate illegal and adverse activity. For instance, in video surveillance, it could indicate personal assault, robbery, and infrastructural sabotage. However, it is not a trivial task; the algorithms have to face different problems: the process of cleaning the noise of the trajectories and the extraction of semantic information that involves experimentation and studies transforming raw movements to other kinds of representations [9]. Moreover, the lack of exact metrics to measure the quality of a semantic extractor makes the study of trajectories difficult.

On the other hand, the use of unsupervised learning to anomaly detection is already justified and used in the literature [10]. The supervised approaches to anomaly detection are less practical in some contexts, such as video surveillance applications and automatic motion learning, since labeled training data are not usually available or practical to obtain. That is why techniques are required to learn anomaly activity patterns in an unsupervised manner. Training data will often contain anomalies or outliers that are unusual or infrequently occurring. The learning algorithm must adapt to anomalies and must be robust in the presence of noise and occlusion.

This work focuses on comparing two descriptors to anomaly trajectory detection taking morphology as the main feature. Moreover, the proposed methodologies present experiments to verify that descriptors improve the trajectory analysis process. These experiments will be validated throughout performance comparisons considering some data sets of the literature. Specifically, we aim to analyze automatic trajectory anomaly detection performance based on unsupervised learning, taking Discrete Fourier Transform and Multilevel Discrete Wavelet Decomposition as principal descriptors. Moreover, we applied our methodology over a real video surveillance data set to identify rare videos based on anomaly trajectories. In summary, our contributions are:

- A methodology to identify anomaly trajectory detection based on Fourier and Wavelet transforms as descriptors.
- Verify the unsupervised learning method as the affinity propagation in the trajectory anomaly detection based on similarity analysis.
- A set of comparative studies revealing interesting patterns about trajectories considering different data sets and descriptors.
- A case study based on real data that demonstrate the usefulness of our methodology to anomaly trajectory

detection on video surveillance.

II. Related works

The literature about trajectory analysis is extensive. To better contextualize our approach, we divide this section into some parts (also considering the datasets used for each analysis): introduction to trajectory analysis, trajectory processing, and anomaly detection.

Kong et al. [11] classified trajectory data as explicit and implicit based on their continuity and structure. Explicit trajectory data provide time and location information, besides a well-structured and Spatio-temporal solid continuity. Trajectories generated by GPS data are the most representative ones in this category. On the other hand, implicit trajectory data has weak spatiotemporal continuity, sub-categorizing this class in signal-based, sensor-based, and network-based data. This study introduces a summary of applications and services that use trajectories, presents an application-based trajectory classification, and also mentions some recommendation system services that use trajectories in their studies. In order to contextualize, according to this study, our data are within the subcategory of sensor-based data since our case study is related to the monitoring of people.

Depending on the entity that originates the trajectories, they will be subjected to a finite set of classes or trajectory types, from regular movements to highly erratic movements. Trajectory modeling is the first and challenging step in the treatment of trajectories. Into the literature, there are different ways of treating trajectories; for instance, the algorithm called TRACLUS [12] processes the trajectories using segments, creating with this information a summary of them. On the other hand, some studies restrict trajectories to a road network; it refers to the movement of vehicles following a transport network. In this category, NETSCAN [12], NNCluster [13], and NEAT [14] show studies restricting the trajectories to a road network. In addition to being restricted to a road network, NETSCAN and TRACLUS process trajectories from the segmentation approach without considering characteristics or patterns that are repeated in different parts of the trajectory (high-level features). Instead, [10], [15] and [16] model trajectories trying to capture information present in the whole way (individual identity).

In this work, to detect the anomaly in feature space, we use the Distance-Based Methods [17]. The trajectories with a long distance from most trajectories are regarded as abnormal, using clustering to create groups of similar trajectories.

Piciarelli et al. [18] created an algorithm to generate synthetic trajectory data sets. This algorithm generates a thousand subsets that are automatically generated with sixteen points each. These data sets are used in more academic papers since this study is one of the first works that address detecting anomalous trajectories and share their data sets. Years later, Laxhammar et al. [8] presented new results considering the algorithms and the datasets generated by Piciarelli. This study emphasized the sequential analysis of incomplete trajectories, which the author termed real-time learning based on an incremental update of the training set.

Ergezer et al. [19] presented a trajectory descriptor with a covariance matrix to detect anomalous trajectories using Nearest Neighbors (NN) and Space Representation (SR), besides the use of spectral grouping for the perception of activity. This study also uses the synthetic data set of Piciarelli [18] as part of their results and also performs experiments with real data of the University of California San Diego (UCSD) and the MIT Parking Lot [20] for the detection of anomalies in videos. In the same vein, Sillito et al. [21] proposed a new framework to detect abnormal trajectories considering the behavior of passersby in terms of trajectory movement. This framework builds a One-Class classifier that is based on probabilities using the Gaussian distribution. Moreover, they conduct experiments using labeled and unlabeled data sets using two databases such as CAVIAR INRIA [22] and Capark [23].

This section presents some studies in trajectory anomaly detection, which use various methods to extract features from trajectories. However, the use of Wavelet and Fourier transforms as a descriptor focusing on trajectory shape or morphology does not present deep studies.

III. Background

This section presents essential concepts related to our approach to familiarize the reader with our research topic.

A. Point, trajectory and sub-trajectory

For our study a point p is a tuple (x, y, t), where x and y are the position (latitude and longitude respectively) and t is the time-lapse when the position is collected:

$$p_k = (x_k, y_k, t_k), k \in \mathbb{N} \tag{1}$$

and a list of points ordered in time forms a trajectory T_i:

$$T_i = (tid_i, \{p_k\}_{k=1:K}) \tag{2}$$

where tid_i is the identifier with $t_1 < t_2 < t_3 < ... < t_K$ in a sequence of points $\{p_1, p_2, ..., p_K\}$ and $\{i, K\} \in \mathbb{N}$.
By the other hand for our study a sub-trajectory is defined as:

$$T'_s = \{p_k\}_{k=1:K} \tag{3}$$

where T'_s is a set of points $p_k = (c_k, t_k)$, t_k is the time instant in which the component c_k is collected and $c_k \in w_x \vee c_k \in w_y$ defined on Equations 10 and 11 respectively.

B. The Discrete Fourier Transform (DFT)

The Fourier transform is a mathematical function that decomposes a waveform, which varies through time, into the frequencies and amplitudes that signal up. The output of the Fourier transform has real and imaginary parts for positive and negative frequencies. The absolute value of their outputs represents the original function frequencies. Thus, the Fourier transforms allowing viewing any function as a sum of simple sinusoids.

The DFT is a type of discrete transformation used in Fourier analysis. The DFT can be defined as the sampling of a function at a certain frequency, and it requires as input a finite discrete sequence. For instance, these sequences can be generated from the sample of a single section of a signal.

Let $f(t)$ be the signal which is the source of the data and let be N instants separated by sample times denoted by $f[0], f[1], f[2], ..., f[N-1]$. The DFT of $f(t)$ can be defined as:

$$F[k] = \sum_{n=0}^{N-1} f[n] W^{kn}, \tag{4}$$

for each $k = 0, 1, \ldots, N-1$. Where $W = e^{-j(2\pi/N)}$ and $j = \sqrt{-1}$ which is an imaginary number. $F[k]$ are the coefficients to each basis function in the linear summation.

C. The Multilevel Discrete Wavelet Decomposition (MDWD)

The Wavelet transform is a mathematical function useful in digital signal processing and image compression. In signal processing, Wavelets make it possible to recover weak signals from noisy ones, which is helpful, especially in the processing of X-ray and magnetic-resonance images in medical applications. The Wavelet and Fourier transform represents a signal through as a linear combination of their basic functions, and both of them decompose signals as a superposition of simple units from which the original signals could be reconstructed. The Wavelet Transform decomposes signals into wavelets, and their base functions are compact or finite in time. This feature allows the Wavelet Transform to obtain time information about a signal in addition to frequency information. The Wavelet transform has a window size that varies frequency scale. This technique is advantageous for the analysis of signals containing both discontinuities and soft components. Short high-frequency base functions are needed for discontinuities, while at the same time, long low-frequency ones are needed for the soft components. The Wavelets are a class of functions used to localize a given function in both space and scaling.

To analyze non-stationary signals, we need to decompose signals into localized units in both time and frequency domains. For this purpose, we use the MDWD. According to [24], the MDWD is a wavelet-based discrete signal method, which can extract multilevel time-frequency features from a signal by decomposing it as low and high-frequency sub-signals level by level.

For the next explanation we use bold symbols such as $\mathbf{x}$, $\mathbf{a}$ or $\mathcal{X}$ to denote vectors and not-bold a, x or l to scalars. We denote the input for N samples for a signal as $\mathbf{x} = \{x_0, x_1, \ldots, x_{N-1}\}$, and the low and high sub-signals generated in the i-th level as $x^l(i)$ and $x^h(i)$. In the $(i+1)$-th level, MDWD uses a low pass filter $\mathbf{l} = \{l_1, \ldots, l_k, \ldots, l_K\}$ and a high pass filter $\mathbf{h} = \{h_1, \ldots, h_k, \ldots, h_K\}$, $K \ll N$, to convolute low frequency sub-signals of the upper level as

$$a_n^l(i+1) = \sum_{k=1}^{K} x_{n+k-1}^l(i) \cdot l_k, \tag{5}$$

$$a_n^h(i+1) = \sum_{k=1}^{K} x_{n+k-1}^l(i) \cdot h_k, \tag{6}$$

where $x_n^l(i)$ is the n-th element of the low frequency signal in the i-th level, and $\mathbf{x}^l(0)$ is set as the input signal. The low and high frequency sub-signal $\mathbf{x}^l(i)$ and $\mathbf{x}^h(i)$ in the level i are generated from the $1/2$ down-sampling of the intermediate variable signals defined as (7) and (8).

$$\mathbf{a}^l(i) = \left\{a_1^l(i), a_2^l(i), \ldots\right\} \tag{7}$$

$$\mathbf{a}^h(i) = \left\{a_1^h(i), a_2^h(i), \ldots\right\} \tag{8}$$

The sub-signals set:

$$\mathcal{X}(i) = \left\{\mathbf{x}^h(1), \mathbf{x}^h(2), \ldots, \mathbf{x}^h(i), \mathbf{x}^l(i)\right\} \tag{9}$$

It is called the i-th level decomposed of $\mathbf{x}$, and it has different time and frequency resolutions. The sub-signals with different frequencies in $\mathcal{X}$ are defined as the MDWD, and it maintains the same order information with the original signal $\mathbf{x}$, and the frequency from $\mathbf{x}^h(1)$ to $\mathbf{x}^l(i)$ is from high to low.

Fig. 1 shows a recreation of MDWD of $\mathbf{x}$ with three levels, each of the pointed lines repair each level, the first convolution with the initial signal is considering as level 0. Each of the rectangles represent the low or high pass filter giving as a result $\mathbf{a}^l(i)$ and $\mathbf{a}^h(i)$, while $\mathbf{x}^l(i)$ represent the input signal to the next level i and $\mathbf{x}^h(i)$ is added to the solution. The components rounded by lines dotted in red make up the decomposition of the original signal in several levels which for this study will be used as features. Finally as a result we obtain $\mathcal{X}(3)$, which in this case it is composed by four sub-signals.

IV. Anomaly Trajectory Detection

In this section, we describe details about the procedures involved in our pipeline. The most important task of our proposed methodology is the description of trajectories; we focus on describe trajectories based on their morphologies for grouping similar ones. Fig. 2 illustrates each of the steps

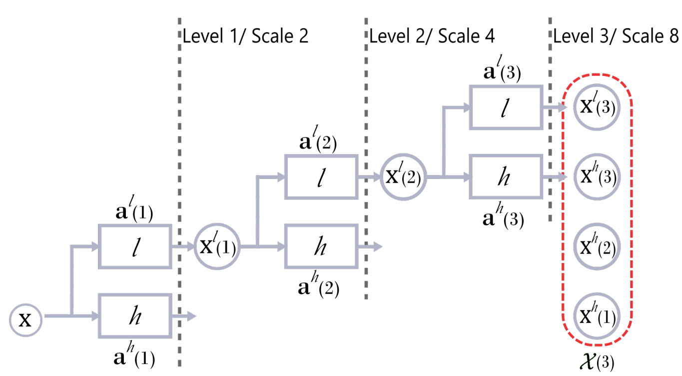

Fig. 1: Representative illustration of MDWD to $\mathbf{x}$ with three levels obtaining as results $\mathcal{X}(3)$. This image is based on [24].

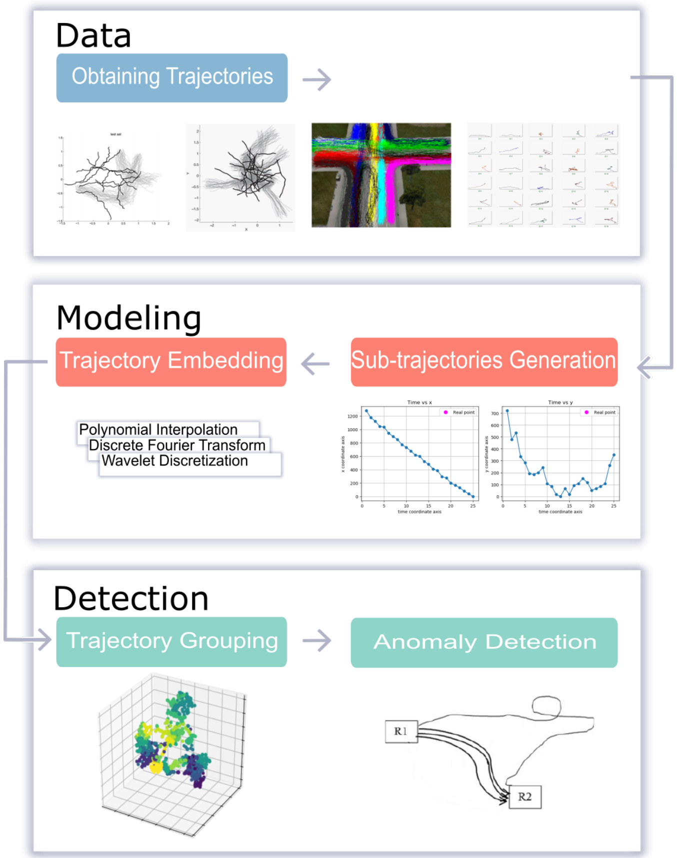

Fig. 2: Pipeline overview of the proposed methodology based on three modules: Data, Modeling, and anomaly detection.

applied in our approach. We divide our methodology into three modules: Pre-processing, Modeling, and Detection.

The pre-processing starts from obtaining trajectories, where sometimes it is necessary to use trajectory transformation algorithms or data cleaning methods. In the real world, trajectories are noisy, and the data are not standardized. The trajectory modeling consists of finding an adequate representation, this representation highlight characteristic that helps to discriminate or classify our data for a specific purpose. The feature extraction is present in this step. Finally, in the third module, the anomaly detection, properly this step consists of detecting an isolated point in a hyperplane since to detect anomalies, in our study, we will use the distance-based methods approach. The trajectories with a long distance from most of them are regarded as abnormal and clustering to create similar groups.

A. Trajectory data

The module Data is about the obtaining and the used trajectories in our study. For our approach, a trajectory represents an object in motion, it having as extremes the beginning and at the end. In this work, we used four datasets:

- The Synthetic dataset created in [18], it is about $260,000$ trajectories generated by an algorithm. These trajectories were divided into $1,000$ groups, and each group contains 260 instances with coordinates (x, y), which 250 belong to 5 clusters, and the last 10 are anomalous trajectories. These trajectories are 16 points long, with no time information.
- The dataset created by Laxhammar et al. [8] using the Piciarelli's [18] algorithm. This new dataset is about $200,000$ thousand trajectories, with 100 groups from 10 different clusters; each group contains $2,000$ trajectories. This dataset allows better tests in efficiency and effectiveness.
- The CROSS [5] dataset contains $9,700$ trajectories simulating four-way traffic intersections with various through and turns patterns, including even a u-turn. The dataset consists of $9,500$ activity paths belong to 19 clusters and 200 anomaly paths. These paths have different lengths with no time information.
- A dataset with $1,000$ trajectories with coordinates (x, y). It contains 970 normal and 30 anomaly paths. These paths have different lengths with no time information. These trajectories were extracted from videos that belong to a Laboratory. The videos are used to analyze anomalous events in simple situations [25], the content is real without forcing any abnormal situation.

All of these datasets are normalized and cleaned; therefore, no pre-processing task was made over them. In this section, it is important to mention that our approach supports trajectories of different sizes; it is corroborated with the experimentation of the Laboratory and CROSS datasets that contain trajectories with different lengths.

B. Trajectory modeling

Next, we will proceed to describe the modeling of trajectories in detail. From once the data sets have been obtained to the generation of feature vectors.

*1) **Trajectory normalization**:* This process was applied for each trajectory. For this purpose, we use the *feature scaling* method.

Let the following spaces be:

$$w_x = \{x_i \in p_i \mid \forall p_i \in T_j\}, \tag{10}$$

$$w_y = \{y_i \in p_i \mid \forall p_i \in T_j\}, \tag{11}$$

where p_i is a point and T_j a trajectory. For all variables x_i and y_i of T_j, the *feature scaling* formulas are applied:

$$x'_i = \frac{x_i - min(w_x)}{max(w_x) - min(w_x)}, \tag{12}$$

$$y'_i = \frac{y_i - min(w_y)}{max(w_y) - min(w_y)}, \tag{13}$$

where min and max return the minimum and the maximum values of a space respectively. After computing each component w_x and w_y with (12) and (13), each element has a new assigned value. We can assign the value zero for the minimum and one for the maximum, and the rest of the intermediate values are scales between those thresholds. For instance, for visualization purposes, we multiplied as the maximum values the dimensions of 720 and 1280 for each component respectively (x and y), obtaining; as a result, the Fig. 3.b (Fig. 3.a shows the original trajectory).

*2) **Trajectory decomposition**:* In the proposed study, the representation of trajectories are generated by splitting them into 1-D sub-trajectories for x and y spaces, represented as $X = x_i, Y = y_i, i = 1, \ldots, n$ (n is the number of points of a trajectory). X and Y represent the horizontal and vertical movements. Fig. 4 shows an example of two sub-trajectories generated by our modeling. Another interpretation that fits into this process is that these sub-trajectories can behave like some time series representing the variation of each spatial component. Thus, the signals give a parameterized behavior compared to trajectory data, and it aids the description in shape.

*3) **Feature space representation**:* Once the *trajectory decomposition* is finished, we can apply feature extraction methods to describe each sub-trajectory. The descriptor inputs are the two sub-trajectories. We consider two techniques to achieve sub-trajectories description, including discrete Fourier transform and MDWD. The derivation of our feature space representation of sub-trajectories using the three proposed methods is specified as follows:

*a) **Discrete Fourier transform**:* The Feature space representation of a trajectory using DFT is similar. The N-points DFT of X (see Section III), defined as a sequence X_f of N complex numbers ($f = 0, ..., N-1$), is given by:

$$X_f = DFT(X) \tag{14}$$

$$Y_f = DFT(Y) \tag{15}$$

X_f and Y_f are complex numbers with the exception of X_0, Y_0 which are real. As a rule, the DFT sequence is truncated after m terms for X_f and k for Y_f. Formally, let be a_i and $\hat{a}_i$ be the real and imaginary part of X_f, and b_i and $\hat{b}_i$ be the real and imaginary part of Y_f. Since we define working with real instead imaginary numbers, we convert X_f and Y_f into real numbers using (16) and (17) respectively.

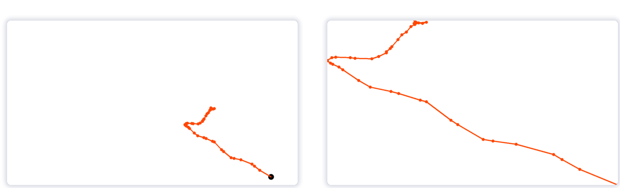
(a) (b)

Fig. 3: The normalization of trajectories improves our features. (a) The initial trajectory corresponds to the path generated in a video segment. (b) Each component of the trajectory has been normalized by video dimensions in each component.

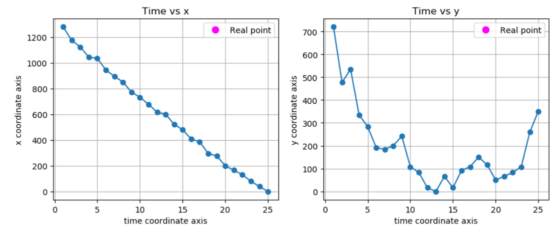

Fig. 4: Trajectory decomposition. Our modeling breaks the trajectories down into two 1-D sets of points.

$$r_i = \sqrt{{a_i}^2 + \hat{a}_i^2}, i = 0, ..., m-1 \tag{16}$$

$$\bar{r}_j = \sqrt{{b_i}^2 + \hat{b}_i^2}, j = 0, ..., k-1 \tag{17}$$

with r_i and $\bar{r}_j$ numbers, we note that some of them appear twice, choosing only one as in (18) and (19).

$$R_x = \{r_0, ..., r_i, ..., r_{m-1}\}_{\neq} \tag{18}$$

$$R_y = \{\bar{r}_0, ..., \bar{r}_j, ..., \bar{r}_{k-1}\}_{\neq} \tag{19}$$

where R_x and R_y are set formed by unique elements. We perform discretization or binning with those two sets of variables, transforming the variable length of a set into a constant defined length set. It was using histograms b_q and b'_q. They meet the following conditions:

$$|R_x| = \sum_{q=1}^{l} b_q \tag{20}$$

$$|R_y| = \sum_{q=1}^{l} b'_q \tag{21}$$

Where b_q and b'_q are functions to count the numbers of observations that fall into each of the disjoint categories (bins). l is the number of bins, $|R_x|$ and $|R_y|$ are the numbers of observation of R_x and R_y respectively. Finally, the trajectory can be represented in the feature space by $\mathbf{F}_{DFT}$ defined as: (22).

$$\mathbf{F}_{DFT} = [\sum_{q=1}^{l} b_q, \sum_{q=1}^{l} b'_q] \tag{22}$$

*b) **Multilevel Discrete Wavelet Decomposition**:* For our description process with MDWD, the *Haar* family is used. It presents different levels of frequencies depending on the number of different forms present in the trajectory. Applying MDWD in X and Y, we can obtain (23) and (24) respectively.

$$[cA_m, cD_m, cD_{m-1}, \ldots, cD_2, cD_1] = MDWD(X) \tag{23}$$

$$[cA_k, cD_k, cD_{k-1}, \ldots, cD_2, cD_1] = MDWD(Y) \tag{24}$$

The output is a list of coefficients, where m and k denote the maximum useful level of decomposition. Thus, the first element cA_m of the result is the approximation coefficients array, and the following elements $cD_m, \dots, cD_1$ are detailed coefficients arrays.

We define the feature vector F_x as the concatenation of different levels of coefficients obtaining with MDWD for X given by (25). A similar expression can be defined for Y as (26).

$$\mathbf{F}_x = [cA_m, cD_m, cD_{m-1}, \dots, cD_2, cD_1] \tag{25}$$

$$\mathbf{F}_y = [cA_k, cD_k, cD_{k-1}, \dots, cD_2, cD_1] \tag{26}$$

with F_x and F_y, we perform discretization using histograms b_q and b'_q, they meet the following conditions:

$$|F_x| = \sum_{q=1}^{l} b_q \tag{27}$$

$$|F_y| = \sum_{q=1}^{l} b'_q \tag{28}$$

In a similar way as was applied with DFT. Finally, trajectory can be represented in the feature space by $\mathbf{F}$ defined as (29).

$$\mathbf{F} = [m, k, \sum_{q=1}^{l} b_q, \sum_{q=1}^{l} b'_q] \tag{29}$$

C. Anomaly detection

Once the feature vectors were obtained, we perform the anomaly detection. We use the distance-based methods [17], the trajectories with a long distance from most trajectories are regarded as abnormal. For this purpose, we use clustering. We segment and separate trajectory information in the clustering process to detect anomalies (which are located at extremes far from majority groups). As a clustering method, we use *affinity propagation* (AP); this method suits our experiments. Moreover, the AP allows the separation of different trajectories since this clustering method generates more groups than other unsupervised methods. Finally, to *recuperate anomaly trajectories* from clusters, we defined a threshold. It is defined as the maximum number of elements of an anomalous cluster.

V. Results and Discussion

In this section, we describe the results of our experiments. Three synthetic datasets perform the quantitative results. Moreover, we present a case study with a real dataset (see Section VI).

*a) **Experimental Setup**:* The hyper-parameters of our experiments are described as follow. For AP the *preference parameter* is set to the median of the input similarities, and the *damping factor* is set to 0.5, 0.625, and 0.7. In some cases, the maximum number of iteration should be set to one thousand. In the case of histograms, the number of bins is set to ten, and for obtaining the *average accuracy*, we choose the best *threshold* in each trajectory subset experimentation.

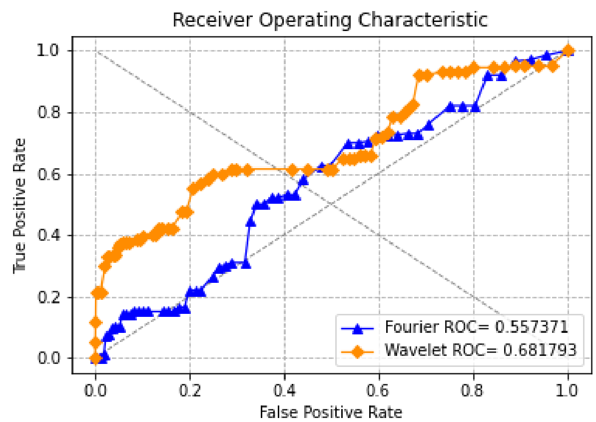

Fig. 5: Comparative evaluation performance of DFT and MDWD descriptors using CROSS dataset.

*b) **Evaluation**:* In order to evaluate the relative performance of our proposed representation in exhaustive datasets, we perform experiments using synthetic datasets generated in [18] and [8]. Due to the nature of these datasets, we use average accuracy to evaluate performance. Instead, with the CROSS dataset, we use accuracy since it is composed of one set of trajectories. The results are present on Table I. We propose to evaluate the performance in this dataset with Receiver Operating Characteristic (ROC) curve, each point of the ROC curve is obtained with a different threshold value, denoting values for True Positive Rate (TPR) and False Positive Rate (FPR) in each detection. This information provides a visual perception of the best threshold using FPR and TPR. We can see our result in Fig. 5. According to it, MDWD descriptors achieve the best results than DFT in this dataset.

*c) **Discussion**:* First, we describe the results achieved with synthetic datasets, and then we explain the results obtained with the CROSS dataset.

For the first two datasets, it is clear that DFT has the highest detection performance. On the third dataset, results for DFT were slightly worse compared to MDWD. Although the accuracy obtained for the CROSS dataset decrease in the two ones, it presents a great difficulty in processing since the trajectories have variable lengths.

From Table I, the best score obtained is with Laxhammar dataset and we consider that it is competitive with related works [18] and [19]. On the other hand, in the CROSS dataset, we compare our results with experiments performed by Morris et al. [5], which use the technique of [15], which identified 84% abnormalities with a 10% of false-positive rate. In our

TABLE I: Quantitative results on three synthetic datasets.

	Datasets		
Method	Piciarelli	Laxhammar	CROSS
MDWD	0.9519	0.9780	**0.8884**
DFT	**0.9525**	**0.9848**	0.8825

case using TPR, we obtain 64% with a 24% false positive rate. Obtaining promising results, since our representations take information related to morphology and the CROSS dataset collects information of shape in their anomaly definition, we consider that this dataset collects similar information to the proposed objectives.

VI. Case Study

To assess the performance of our approach, we conduct a case study to identify rare videos based on anomaly detection of people's trajectories. For that, we used SSIG-dataset filmed in a smart sense laboratory door. This dataset contains people in different situations: pointing in and pointing out the laboratory, closing and opening the door, stopping and walking outside the laboratory. The criteria for defining "normal" videos are: entering and leaving the laboratory by opening and closing the door; a short amount of time spent in front of the camera (less than 10 seconds); people going through the corridor outside the laboratory; and people leaving and entering the side laboratory. On the other hand, the criteria to define rare behavior are: making several movements to come and go to the laboratory, stand in front of the camera for a long period, and using the key-box located near the door for an extended period.

The dataset consists of $5,025$ videos which last from two seconds to four minutes and twenty-three seconds recorded during two years. For each video, based on *post-estimation*, we selected a *fiducial point* of a person, and with *tracking* method, we connected the *fiducial point* of each frame generating a trajectory. Aiming to explore a considerable number of samples, we randomly selected $1,000$ trajectories. The idea behind this data set is to segment a subset that contains abnormal behaviors of a person in terms of their displacement in the video (abnormal trajectories). For that, we manually generate a ground truth considering the conditions for rare and normal behavior.

A. Feature extraction and clustering

The first step to accomplish this case study is feature extraction. In this work, we considered Fourier and Wavelet transforms as trajectory feature extractors. Moreover, for this case study, we considered an additional simple descriptor based on interpolation, in order to have a simple description method to compare. Once the coefficients are found for each trajectory, they are used as a feature vector in the clusterization process. The easiest way to achieve the feature extractor performance is to compare an element with its nearest neighbor. Fig. 6 shows the result of a finding of the nearest neighbor trajectory using Kernel Density Estimation (KDE). The first column illustrates a random example, while the second column shows the most similar trajectory considering the vector of each descriptor. Notice that Fourier and Wavelet Transform extract better similar neighboring trajectories, demonstrating that they pull characteristics for grouping better than the interpolation method. Thus, these are used as descriptors of the morphology of our trajectories.

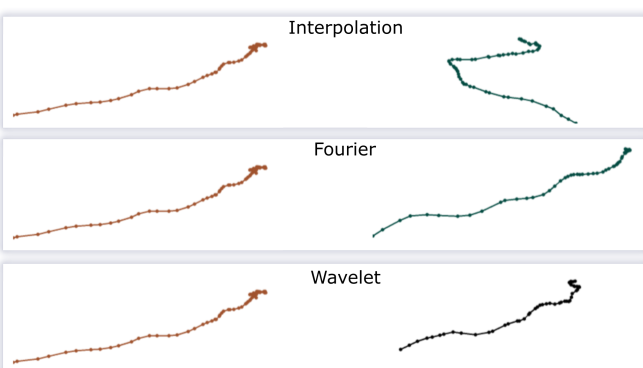

Fig. 6: The most similar trajectory for each descriptor.

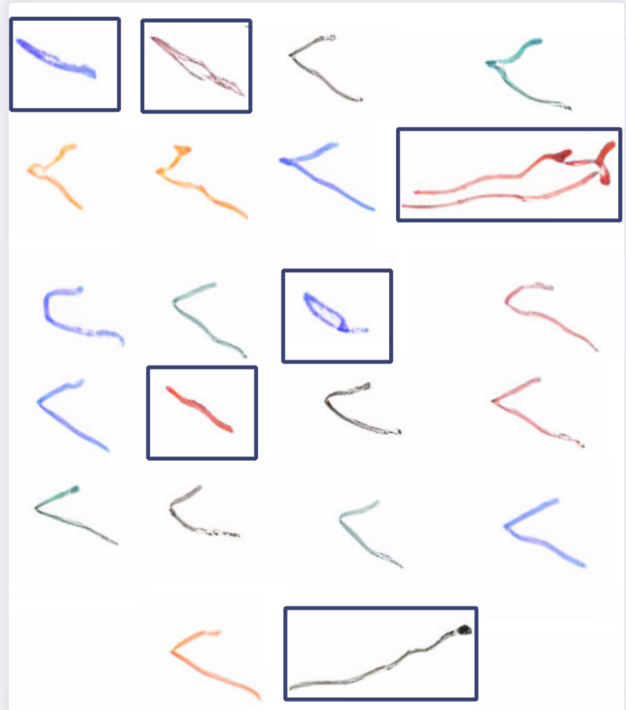

Fig. 7: Example of trajectories of a cluster. The boxed trajectories are wrong clustered.

In the next step, using *Affinity Propagation*, we clustered our trajectories based on their morphology. To measure the performance of our clustering method, we used the *counting* method. In each cluster, this metric counts how many elements are wrong clustered. The average error is computed based on the number of elements of the cluster and the number of wrong clustered elements. For instance, Fig. 7 shows the trajectories of a cluster; the boxed trajectories are wrong clustered. In this example, the cluster has 22 elements, with six wrong clustered trajectories, having 27.27% error percentage (e_i). The total error percentage is calculated by the average of the percentage of each cluster ($E = \sum^n e_i/n$).

TABLE II: Error percentages for each descriptor.

Descriptor	*Affinity Propagation* ($E\%$)
Interpolation	15.67
Fourier	9.20
Wavelet	**6.77**

TABLE III: Results of gathering anomaly trajectories by our choice.

Threshold	*Accuracy*	*Precision*	*Recall*	*Specificity*
1	0.99	1.00	0.70	1.00
2	**0.99**	**0.97**	**0.93**	**0.99**
3	0.989	0.731	1.00	0.98

In order to explore each feature extractor method, we conduct an empirical analysis examining the error percentage of each method. Table II shows the result of the clustering error percentage of each method. Notice that the Wavelet descriptor has the lowest error percentage, following by Fourier.

Based on the lowest error percentage of Table II, in this case study, we use Wavelet transform as the main descriptor. In order to validate our results, it is necessary to set a *threshold* which define as the maximum number of elements that a cluster can have to be considered abnormal. Table III shows the threshold influence in each quality metric. These metrics were computed considering the ground truth. We can see that 2 is the best threshold with 0.97 average for each metric. Note that *Accuracy* values are more significant than 0.989, most of them with 0.99, showing the good performance of our approach.

We present some visual results regarding the qualitative evaluation of our choice (wavelet, Affinity propagation, and threshold 2). Fig. 9 shows anomaly trajectories detected by our approach contained in the rare videos of the ground truth. Fig. 10 shows three clusters generated by our choice. We can see that each cluster groups similar trajectories by their morphology. Finally, Fig. 8 shows two thumbnails after our processing on surveillance videos. The trajectory is represented on the video, each point represent the position in each frame while the white point represents the point taken for the generation of the trajectory. This point allows the observation of the direction that the trajectory takes on each detected point. It could be noticed that the abnormal trajectory presents pronounced deformations (see Fig. 8) while the normal trajectory has smooth chances (see Fig. 10).

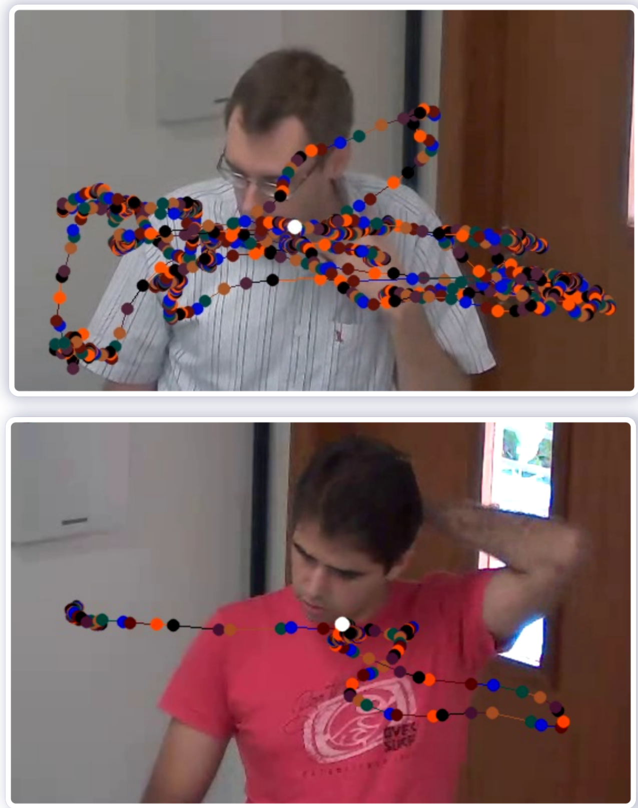

Fig. 8: Thumbnails of videos with their respective trajectories. Rare videos with the abnormal trajectories (pronounced deformation). The white point represents the *fiducial point* taken as a reference, while the colored points represent the *fiducial point* in each video frame.

VII. Conclusión

This paper presents a comparative analysis of trajectory descriptors using coefficient feature space representation to detect anomaly trajectories. We have also introduced the MDWD as a shape-feature extractor on trajectories, and it yields satisfactory results compared to other descriptors, obtaining greater performance in the detection of anomaly trajectories, due to the better trajectory description provided by this method. Our study was based on an unsupervised learning method using similarity analysis—the validation process took into account various synthetic and real-life datasets. Moreover, the usefulness of our approach has been demonstrated throughout a case study to detect anomaly trajectories in real video surveillance data set.

We observe that the used dataset influences the AP algorithm. Whether the number of classes of trajectories increases, the classification precision of the algorithm decrease. There is a possible improvement in the unsupervised learning process using the Adaptive AP method [26] to automatically select the preference parameter and find the optimal clustering solution, and also in the use of k-Nearest Neighbor in order not to define a threshold manually and localize the anomalies automatically on feature space.

References

[1] M. J. Wisdom, N. J. Cimon, B. K. Johnson, E. O. Garton, and J. W. Thomas, "Spatial partitioning by mule deer and elk in relation to traffic." in *In: Transactions of the 69th North American Wildlife and Natural Resources Conference: 509-530*, 2004.

[2] M. D. Powell and S. D. Aberson, "Accuracy of united states tropical cyclone landfall forecasts in the atlantic basin (1976–2000)," *Bulletin of the American Meteorological Society*, vol. 82, no. 12, pp. 2749–2768, 2001.

[3] Y. Pang and Y. Liu, "Conditional generative adversarial networks (cgan) for aircraft trajectory prediction considering weather effects," in *AIAA Scitech 2020 Forum*, 2020, p. 1853.

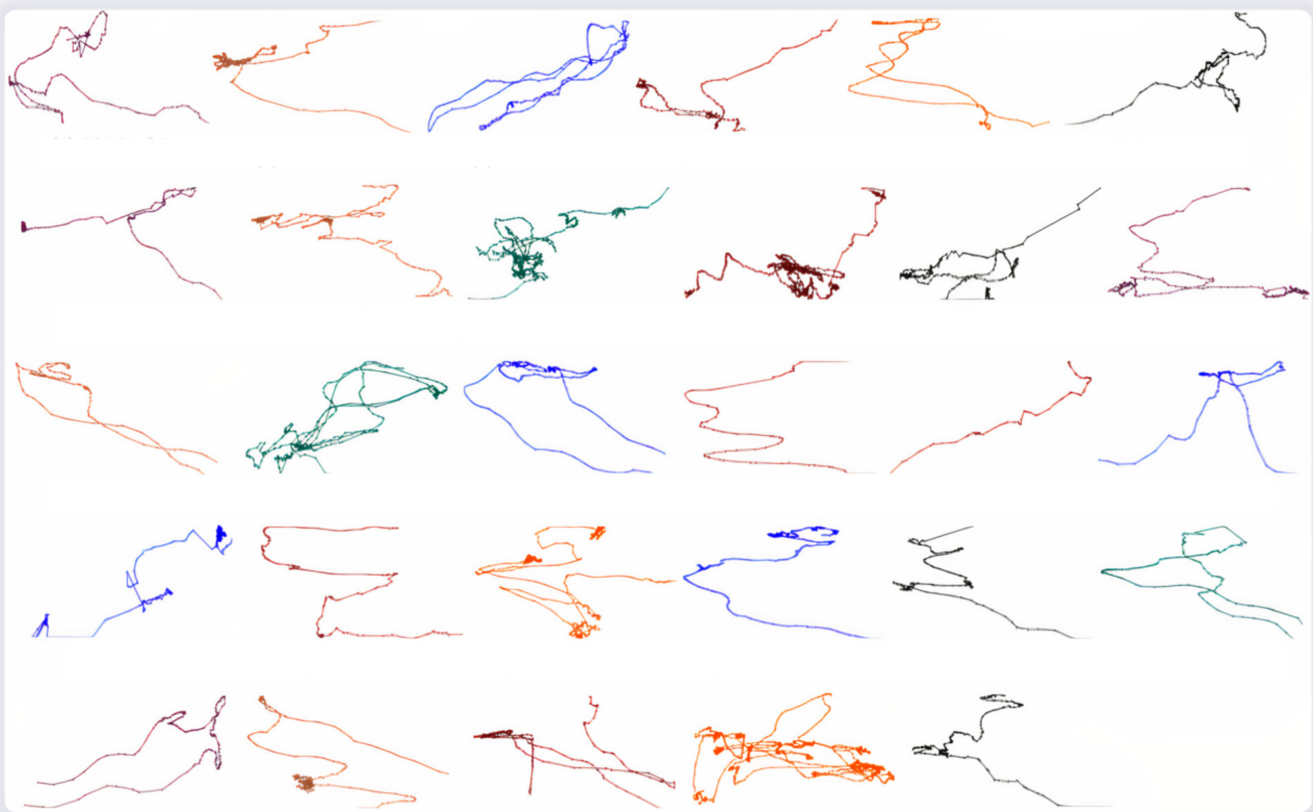

Fig. 9: Trajectories detected as anomaly by our approach.

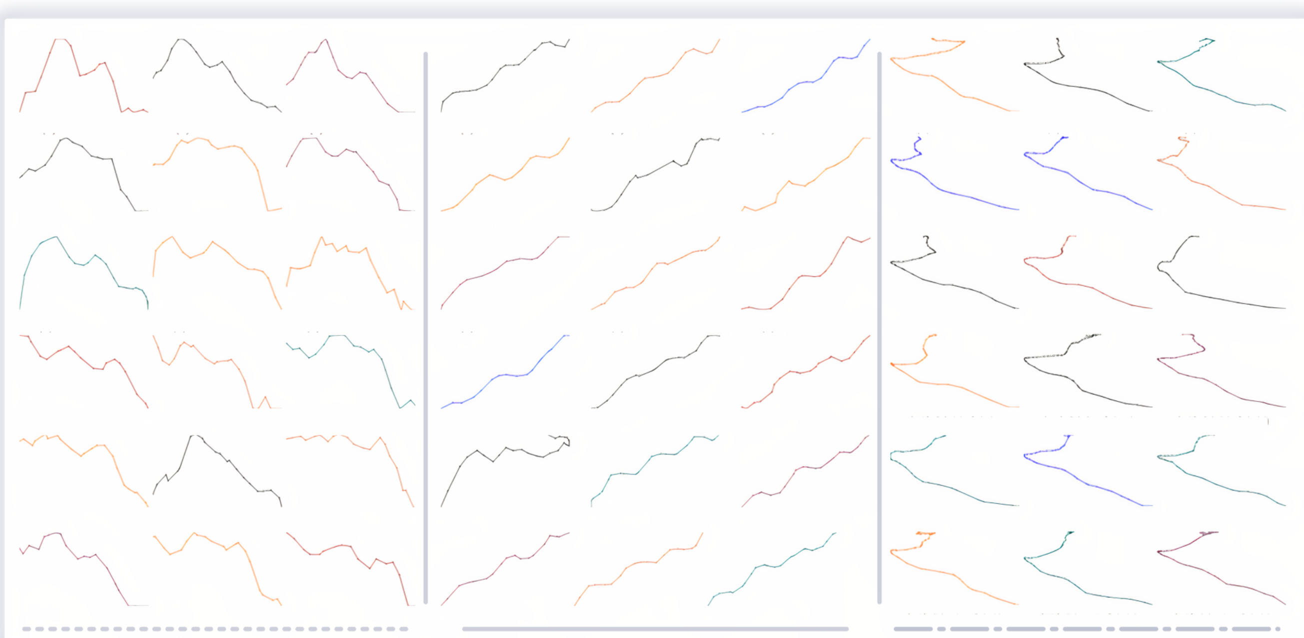

Fig. 10: Normal trajectories detected by our approach and three different generated clusters (columns).

[4] J. Kim and H. S. Mahmassani, "Spatial and temporal characterization of travel patterns in a traffic network using vehicle trajectories," *Transportation Research Procedia*, vol. 9, pp. 164–184, 2015.

[5] B. T. Morris and M. M. Trivedi, "Trajectory learning for activity understanding: Unsupervised, multilevel, and long-term adaptive approach," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 11, pp. 2287–2301, 2011.

[6] F. Turchini, L. Seidenari, and A. Del Bimbo, "Understanding sport activities from correspondences of clustered trajectories," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 43–50.

[7] S. Paul, F. Hole, A. Zytek, and C. A. Varela, "Flight trajectory planning for fixed-wing aircraft in loss of thrust emergencies," *arXiv preprint arXiv:1711.00716*, 2017.

[8] R. Laxhammar and G. Falkman, "Online learning and sequential anomaly detection in trajectories," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 6, pp. 1158–1173, 2014.

[9] C. Parent, S. Spaccapietra, C. Renso, G. Andrienko, N. Andrienko, V. Bogorny, M. L. Damiani, A. Gkoulalas-Divanis, J. Macedo, N. Pelekis *et al.*, "Semantic trajectories modeling and analysis," *ACM Computing Surveys (CSUR)*, vol. 45, no. 4, pp. 1–32, 2013.

[10] S. Khalid and A. Naftel, "Automatic motion learning in the presence of anomalies using coefficient feature space representation of trajectories," *Acta Automatica Sinica*, vol. 36, no. 5, pp. 655–666, 2010.

[11] X. Kong, M. Li, K. Ma, K. Tian, M. Wang, Z. Ning, and F. Xia, "Big trajectory data: A survey of applications and services," *IEEE Access*, vol. 6, pp. 58 295–58 306, 2018.

[12] J.-G. Lee, J. Han, and K.-Y. Whang, "Trajectory clustering: a partition-and-group framework," in *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, 2007, pp. 593–604.

[13] G.-P. Roh and S.-w. Hwang, "Nncluster: An efficient clustering algorithm for road network trajectories," in *International Conference on Database Systems for Advanced Applications*. Springer, 2010, pp. 47–61.

[14] B. Han, L. Liu, and E. Omiecinski, "Neat: Road network aware trajectory clustering," in *2012 IEEE 32nd International Conference on Distributed Computing Systems*. IEEE, 2012, pp. 142–151.

[15] A. Naftel and S. Khalid, "Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space," *Multimedia Systems*, vol. 12, no. 3, pp. 227–238, 2006.

[16] R. Annoni and C. H. Forster, "Analysis of aircraft trajectories using fourier descriptors and kernel density estimation," in *2012 15th International IEEE Conference on Intelligent Transportation Systems*. IEEE, 2012, pp. 1441–1446.

[17] H. Zhang, Y. Luo, Q. Yu, L. Sun, X. Li, and Z. Sun, "A framework of abnormal behavior detection and classification based on big trajectory data for mobile networks," *Security and Communication Networks*, vol. 2020, 2020.

[18] C. Piciarelli, C. Micheloni, and G. L. Foresti, "Trajectory-based anomalous event detection," *IEEE Transactions on Circuits and Systems for video Technology*, vol. 18, no. 11, pp. 1544–1554, 2008.

[19] H. Ergezer and K. Leblebicioğlu, "Anomaly detection and activity perception using covariance descriptor for trajectories," in *European Conference on Computer Vision*. Springer, 2016, pp. 728–742.

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

[21] R. R. Sillito and R. B. Fisher, "Semi-supervised learning for anomalous trajectory detection." in *BMVC*, vol. 1, 2008, pp. 035–1.

[22] I. Labs. (2004) Caviar "inria" dataset. [Online]. Available: https://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/

[23] H. M. Dee and D. C. Hogg, "Detecting inexplicable behaviour." in *BMVC*. Citeseer, 2004, pp. 1–10.

[24] J. Wang, Z. Wang, J. Li, and J. Wu, "Multilevel wavelet decomposition network for interpretable time series analysis," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2437–2446.

[25] R. Mora, E. Cayllahua, V. C. de Melo, G. Cámara-Chávez, and W. R. Schwartz, "Anomaly event detection based on people trajectories for surveillance videos," in *VISIGRAPP - 15th International Conference on Computer Vision Theory and Applications*, 2020, pp. 107–116.

[26] K. Wang, J. Zhang, D. Li, X. Zhang, and T. Guo, "Adaptive affinity propagation clustering," *arXiv preprint arXiv:0805.1096*, 2008.