Identifying Loitering Behavior with Trajectory Analysis

Johnny Núñez[1,2]
jnunezca11@alumnes.ub.edu

Zenjie Li[2]
zli@milestone.dk

Sergio Escalera[1,3]
sescalera@ub.edu

Kamal Nasrollahi[2,3]
kn@create.aau.dk

Universitat de Barcelona and Computer Vision Center [1]

Milestone Systems[2]

Aalborg Universitet[3]

Abstract

The act of remaining in a public area for an extended period is commonly referred to as Loitering, and it is often viewed as suspicious activity with regard to public safety. The research landscape on loitering detection is diverse, featuring various definitions and methodologies. This lack of standardization in defining loitering hamper the generalizability of detection methods. Our work, focuses on providing a clear definition of loitering and detecting it through trajectory analysis. We enrich the field of loitering detection research by introducing a dataset with annotated loitering behaviors. Our contribution is to annotate loitering behavior in the Long-term Thermal Drift Dataset, which already complies with privacy standards. The dataset features a variety of loitering behaviors observed through a real-world thermal surveillance camera across different environmental scenarios. To identify loitering behavior, we employ trajectory analysis methods. These methods quantify parameters such as movement directionality, pace, and dwell time, providing fundamental aspects for loitering detection studies. The dataset and the code are available on https://github.com/johnnynunez/RS-WACV24_Loitering.

1. Introduction

Loitering detection in intelligent surveillance systems has become popular due to the increasing demand for security and safety in public spaces. Loitering is to remain in an area without obvious purpose. Extended stays in public areas are often considered suspicious and can indicate a potential threat to public safety. Some research has been conducted on the detection of loitering [4, 6]. However, the literature presents varying definitions of the term, highlighting the complexity of programmatically distinguishing it from benign activities. This lack of a unified definition is compounded by the diversity of behaviors and interpretations that the term can encompass. Consequently, the absence of a single, common definition makes it challenging to develop robust algorithms that can effectively identify or differentiate this term from other activities.

The cornerstone of our research lies in the development of a publicly available loitering dataset, which is derived from the Long-term Thermal Drift Dataset [17]. With this dataset we define what loitering is in order to provide a comprehensive range of loitering behaviors captured by a real-world thermal surveillance camera in various environmental conditions, including clear skies, rain, snow, and mist. Through our manual annotations, our loitering dataset addresses a critical gap in loitering detection research, presenting a diverse set of scenarios for validating loitering detection algorithms without privacy issues.

With the newly provided data and using the proposed definitions of loitering, we employ trajectory analysis methods for the automatic detection of these loitering behaviors [8]. These methods are designed to quantify parameters such as movement directionality, area and dwell time, which are critical for distinguishing loitering from other activities.

Given our definitions of loitering and the use of trajectories as a feature, our approach is applicable to other data sets when calibrated trajectories are obtained, e.g. using normalized or world coordinates trajectories. Moreover, we show how our trajectory analysis techniques allow for unsupervised application. Additionally, taking into account the provided annotations, we also test the performance of our descriptors using supervised learning.

Our main contributions in the field of loitering detection are:

- We create the largest annotated loitering detection dataset up to date to our best knowledge, derived from the Long-term Thermal Drift Dataset [17]. Our publicly available dataset avoids privacy concerns through the use of thermal data.
- We establish a definition of loitering scenarios, which serves as the basis for subsequent analyses and validations.

- We present a set of simple and efficient baselines based on trajectory analysis on the provided data, and test the computed descriptors both in unsupervised and supervised scenarios.

2. Related Work

Prior works on appearance-based algorithms for anomaly detection, including loitering behaviors, extract complex features from images, thus providing nuanced scene understanding [3, 23, 26]. These algorithms incorporate temporal aspects to comprehend long-term behaviors but usually require labor-intensive detailed annotations for supervised learning. Additionally, they struggle with low-quality sensor data. Notably, methods like [23] often utilize 3D convolutional neural networks in conjunction with generative adversarial networks and autoencoders. Other works such as [26] and [3] adopt transformer architectures for video anomaly identification.

Contrary to previous approaches, another trend in the literature focuses on tracking and trajectory analysis with unsupervised methods for the detection of loitering [4, 8]. The construction of a trajectory typically involves tracking a pedestrian through multiple frames of a video sequence with detectors [9]. These trajectories are then analyzed to detect loitering and to understand trajectory loitering patterns over time, making them less susceptible to image noise compared to appearance-based approaches. Prior studies, such as [8], introduce methods for extracting pedestrian trajectories. These methods evaluate loitering behavior by examining two key parameters: the time duration and the angular change between the starting point and subsequent points along the object's path. By focusing on these metrics, the approach differentiates loitering from regular movement. Similarly, on [24] authors perform feature extraction based on trajectory, BLOB area, and velocity. [4] constructs a 3D virtual space that includes the ground plane and a plane at the pedestrian's head level. This 3D space is used to analyze pedestrian movement more accurately than traditional 2D methods. Authors employ circular variance to discriminate abnormal behavior, which is calculated based on the direction angles of pedestrian trajectories. In [22], dense trajectory descriptors are used to capture the characteristics of human walk. Frame differencing and wavelet transform techniques are combined to identify moving objects, referred to as blobs. These blobs are subsequently classified using a Support Vector Machine (SVM). For tracking individuals across frames, features such as clothing color and texture are also considered. Similarly, [6] employs dense trajectory descriptors to represent human walk parameters, focusing on capturing individual movement patterns. Also, Grid-based techniques, as demonstrated in [5], are useful to quantify behavioral patterns by capturing microscopic randomness in people's traffic lines, which might imply analyzing individuals trajectories.

Most of previous works are based on unsupervised learning, with no explicit labeling and training on the loitering class. However, the use of machine learning techniques has shown potential for trajectory analysis in loitering detection [15, 18, 27]. In this case, labeled data are crucial for the models to effectively differentiate loitering from other types of behavior in surveillance footage. Therefore, the absence of temporal complexities is offset by the need for high-quality annotated data for model training. In [27] they define trajectory redundancy as a quantitative measure for the characteristics of ship loitering. The innovative is that they use a multi-scale sliding window-based method. It is designed for detecting ship loitering across different spatial ranges and time durations. Additionally, a Convolutional Neural Network (CNN) model is trained to identify four typical shapes of loitering trajectories. In the work by [16], loitering detection relies on two steps. First, the YOLOv3 [21] algorithm identifies individuals in video. Then, the DeepSORT algorithm [25] is employed to monitor the individual's movement within the video. Subsequently, the path followed by the individual is scrutinized by comparing both the time spent and distance covered against predetermined criteria. Meeting these thresholds leads to a loitering classification, reducing false positives. The approach in [13] integrates detection and tracking algorithms with person identification techniques. This allows for the measurement of the time an individual spends in a designated area and subsequent face capture.

After reviewing existing algorithms, we employ both unsupervised and supervised trajectory analyses for loitering detection on our dataset. The unsupervised approach is adaptable, allowing for effective detection on any dataset if it has calibrated trajectories, e.g. using normalized or world coordinates trajectories. Thanks to our provided annotations, we also demonstrate the benefits of using a supervised approach with the proposed trajectory features. Both approaches have distinct advantages, making our methodology versatile for various surveillance requirements and providing a comprehensive set of baselines on the provided dataset.

3. Datasets

3.1. Comparison with Existing Datasets

Before presenting our dataset, we review the current landscape of datasets in the realm of surveillance video analysis. Several datasets have been widely used for various purposes, including but not limited to anomaly detection, pedestrian tracking, and activity recognition. Here, we review some of the well-known datasets and discuss their benefits and limitations in the context of loitering.

3.1.1 PETS2007

The PETS2007 dataset [1] is one of the oldest and most commonly used datasets in the field of surveillance video analysis. However, it is not available anymore. A new version of the dataset PETS2014 [19] only contains loitering in 3 video sequences, each lasting 60 seconds.

3.1.2 CUHK Avenue Dataset

The CUHK Avenue dataset [11] focuses on anomaly detection in surveillance settings. It includes 16 training and 21 testing video clips, totaling approximately 30,652 frames. All clips are captured from a single camera on the CUHK campus. While the dataset serves its purpose in specific research contexts, it falls short in terms of variability for a broader application in loitering detection through trajectory analysis. Specifically, it contains only 5 instances of loitering, limiting its applicability for comprehensive studies in this area.

3.1.3 UCSD Anomaly Detection Dataset

UCSD anomaly dataset [10] consists of video footage from surveillance cameras at the University of California, San Diego. No details about loitering are provided. It comprises two subsets: Ped1 and Ped2. The primary focus of this dataset is to facilitate the study of anomalies within crowded scenes, particularly in different crowd densities that range from sparse to very crowded scenarios. While extensively used for motion anomaly detection, specifics on loitering incidents are not provided.

3.1.4 Subway Dataset

The Subway dataset [2] contains video footage of subway stations and is mainly used for detecting anomalous events. It offers a different environmental setting (Entrance and Exit labels) and unusual events containing walking in wrong directions and loitering. However, it is restricted to indoor scenarios, limiting its applicability to broader surveillance settings. In terms of loitering instances, the Subway-Exit section only contains 3 instances, while the Subway-Entrance section includes just 14 instances.

3.1.5 ShanghaiTech Campus

The ShanghaiTech Campus dataset [12] comprises 13 scenes featuring varying lighting conditions and camera orientations. While it includes 130 abnormal events and over 270,000 training frames, its primary focus is on general anomaly detection rather than specialized behaviors like loitering. If segmented by scene, each would contain an average of only 10 anomalous events, making it less suitable for granular analysis. Although it may contain specific cases of loitering, related information is not provided.

3.1.6 UMN Unusual Crowd Activity Dataset

The UMN Unusual Crowd Activity dataset [14] from the University of Minnesota is designed for crowd behavior analysis and anomaly detection. The dataset comprises the videos of 11 different scenarios of an escape event in 3 different indoor and outdoor scenes. There is no clear specification of a split between training and testing frames. Anomaly labels are given solely based on frames, not on individual persons. This could limit the assessment of behaviors specific to each person. Additionally, the events in the dataset are staged, which may not accurately represent real-world, spontaneous situations. Furthermore, the dataset does not provide information about the presence of loitering instances.

3.1.7 Street Scene Dataset

The Street Scene dataset [20], consists of 46 training video sequences and 35 testing video sequences taken from a static USB camera looking down on a scene of a two-lane street with bike lanes and pedestrian sidewalks. It focuses on naturalistic driving scenarios. It is mainly employed for object detection and tracking in urban settings. The dataset contains a wide variety of objects and actions but is primarily geared toward vehicular movement. This focus makes it less suitable for analyzing pedestrian behavior, such as loitering. There are only 36 instances of loitering across different test scenarios.

3.2. Rationale for Long-term Thermal Drift Dataset

The Long-term Thermal Drift Dataset [17] is a public dataset containing thermal surveillance imaging from a single location across 8 months, incorporating diverse environmental and pedestrian scenarios. Each video clip lasts for 2 minutes with 1 frame per second. Importantly, it captures real-life conditions rather than staged setups and avoids privacy concerns using a thermal camera. As a result, it offers a reliable basis for evaluating trajectory analysis methods in loitering detection, further discussed in Section 5. While the dataset comes with human-annotated bounding boxes and tracking IDs, it lacks loitering annotations. In this work, we enrich this dataset by providing extensive loitering annotations across a wide variety of trajectory cases, defining the largest dataset up to date in terms of number of annotated loitering samples.

3.3. Our Annotated Dataset

The Long-term Thermal Drift Dataset is already labeled with bounding box centers and tracking IDs. We can define sequences composed of tracked coordinates in the form:

$$s_i = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\},$$

where i is the ID used for pedestrian identification consistent throughout the tracked sequence, $n \in (1, 120)$, and pairs (x, y) represent the coordinates of the center of bounding boxes. These coordinates are transformed using the Inverse Perspective Matrix to mitigate perspective distortion. Based on this dataset of trajectories, next we describe the protocol for annotating loitering cases and the new annotated dataset details.

3.3.1 Definition of Loitering

Loitering is a behavior that has garnered considerable attention in surveillance studies due to its complex and ambiguous nature. Essentially, loitering refers to the act of remaining in a particular location without an obvious reason. However, the challenge in identifying and classifying loitering arises from its varied manifestations and the differing operational definitions across academic and professional communities. In the following, we discuss the main four categories of loitering behavior that we consider in this work, and that group most previous loitering behaviors presented in literature. A visual example for each of those loitering groups is shown in Figure 1 with real cases from the annotated dataset.

- **Seated loitering** (Figure 1a): individuals are found sitting in one area for an extended period. While seemingly innocuous, seated loitering can sometimes be a prelude to other activities that may be of concern in a surveillance context.
- **Random Walk loitering** (Figure 1b): this involves individuals moving around aimlessly within a specific area. The movement pattern lacks a clear trajectory or purpose, which makes it challenging to differentiate from normal pedestrian activity using typical tracking algorithms.
- **Abnormal Trajectory** (Figure 1c): this displays deviations from the common behavior of crossing the port. Individuals may circle a particular object, move in a zigzag fashion or circular movements, making it easier to flag as potential loitering by surveillance systems.
- **No Motion** (Figure 1d): individuals stand still in one location for an extended period.

3.4. Dataset Details

For the loitering annotated version of the dataset, We annotated 31 days in 5-day intervals across the different months of the dataset. The total number of labelled video clips is 1,005. Each video has a duration of 2 minutes, amounting to 120 frames in total. Thus, we have a total of 120,600 annotated frames. The videos have a resolution of 288 $\times$ 384 pixels. We follow strictly the protocol defined in Figure 1 to annotate cases of loitering. Without loss of generality, we assign only one binary label (loitering or NO loitering) to each entire trajectory instance. In total, there are 19,737 IDs annotated (one ID corresponds to one trajectory). From these annotated trajectories, 80.44% correspond to non-loitering, and 19.56% to loitering. We defined the partitions with a training set containing 15789 IDs and a test set containing 3948 IDs, maintaining the same proportion of loitering cases. Because of the camera's position, two major issues arise. The first is occlusion, as a pillar in the middle of the image produces noisy annotations. To address this, a mask generated by SAM [7] removes all the coordinates associated with the pillar. The second problem is that the camera is situated in the perspective of the port. Given that the camera's intrinsic parameters are unknown, we undo perspective applying an Inverse Perspective Matrix (IPM), using several points of interest to calculate and apply the matrix. We use the Savitzky-Golay filter with a window size of 11 and a polynomial order of 3 to smooth trajectories without reducing the number of points, thus preserving the temporal information. In Figure 2, we show the distribution of annotated IDs across the number of frames in which they appear, categorized into loitering and non-loitering cases. In particular, there is a large peak at 120 frames for loitering cases, suggesting that objects or individuals engaging in loitering behavior frequently remain in the scene throughout the entire 120-frame duration of a video clip. This aligns with the notion that loitering involves a level of inactivity or stationary behavior. On the contrary, non-loitering cases display a broader distribution across frame counts, indicating a more dynamic presence in the video clips. Furthermore, there is a relatively lower frequency of loitering cases appearing for shorter frame counts, reinforcing the idea that loitering behavior usually involves longer periods of visibility in the scene.

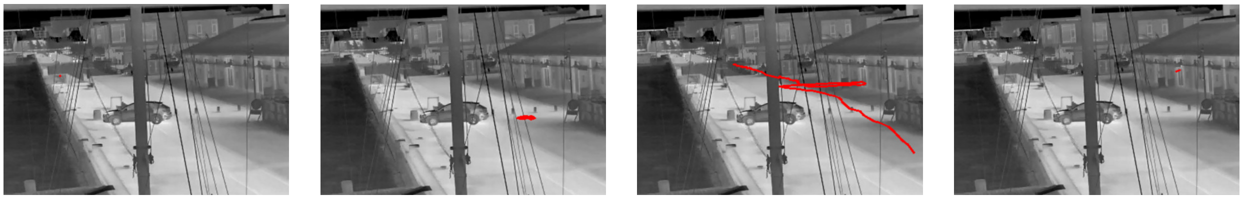
4. Methodology

4.1. Loitering Detection Pipeline

We apply the pipeline shown in Figure 3 to smooth trajectories and apply the geometric analyses. This methodology includes algorithmic streamlining of trajectories, employing distinct algorithms inspired by prior works to discern normal trajectory from loitering behavior.

4.2. Motion Analysis

Motion analysis plays a pivotal role in trajectory examination. Angles between successive direction vectors are computed to identify abrupt variations in direction, instrumental in pinpointing potential loitering intervals within the



(a) (b) (c) (d)

Figure 1. Loitering categories: (a) Seated; (b) Random Walk; (c) Abnormal Trajectory; (d) No Motion. Trajectories are shown in red.

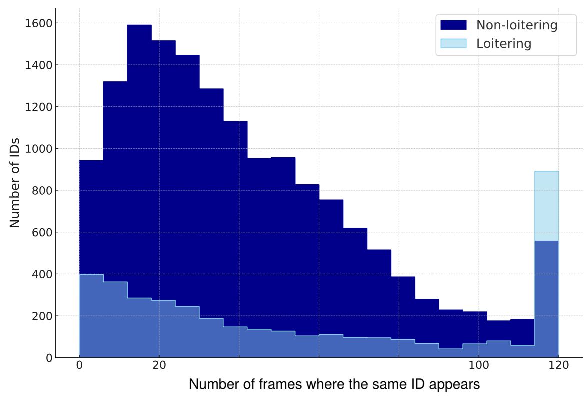


Figure 2. Histograms showing the count of frames in which the same ID appears as either loitering or non-loitering within our loitering dataset. The histogram for loitering is superimposed on the one for non-loitering.

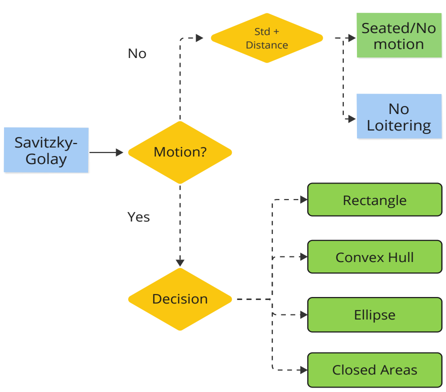


Figure 3. Geometric analysis methodology pipeline.

trajectory. If there are at least 4 or more recognizable points, there is motion. We identify areas of loitering by examining three key parameters within a trajectory: angle θ, number of points, and a sliding window. A low angle θ indicates tight turns in the path, suggesting more complex or erratic movement. The parameter number of points measures how often such turns or directional changes occur; a high count indicates frequent changes. The ‘sliding window’ is a subtrajectory that we analyze within the larger path to pinpoint specific segments where loitering might be taking place. These three parameters together provide a comprehensive way to detect motion and potential loitering areas.

4.3. Stationarity Analysis

To detect seated loitering Figure 4a we create two concepts. In short-term, we iterate through the trajectory in segments defined by a sliding window. For each segment, we calculate the geometric center, assess the distances of all points in the segment to this center, and check whether the conditions for loitering (all distances within radius and standard deviation below threshold) are met. The iteration step is half the frame threshold, allowing for overlapping segments and robust loitering detection. In long-term, we compute the geometric center of the entire trajectory and evaluate the distances of all points in the trajectory to this center. If all distances are within the specified radius, and the standard deviation of these distances is below a threshold, loitering is detected.

4.4. Geometric Analyses

Different geometric analyses are employed to ascertain loitering:

- **Rectangle Area Method:** The algorithm examines spatial confinement within a rectangular area (Figure 4b) over a specific trajectory duration. Critical variables include four points from the motion algorithm and a time threshold T_0. Additionally, an area threshold A_0 smallest rectangle’s area indicates loitering. This analysis represents confined motion, indicating Random Walk loitering or small areas of loitering.

- **Convex Hull Method:** The algorithm computes the smallest convex set that encapsulates all given trajectory points (Figure 4c). Key variables include the area enclosed by the convex hull and an area threshold A_1 for comparison. Loitering is detected if the hull area is below A_1. A flatten hull area indicates that the trajectory is normal, and a sparse hull area indicates a big area of movement. This method is suitable for identifying any kind of loitering.

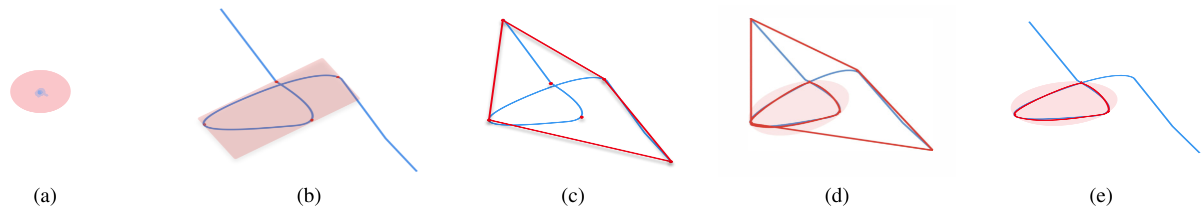


Figure 4. Trajectory Analysis Methods. (a) No Motion (b) Rectangle. (c) Convex Hull. (d) Convex Hull + Ellipse. (e) Closed Area.

- **Convex Hull + Ellipse Fitting:** This method builds on top of the Convex Hull Method by fitting an ellipse within the hull points (Figure 4d). Critical variables are the dimensions of the fitted ellipse, the hull area, and two thresholds: an area threshold A_2 and an ellipse fitting precision threshold P_0. The method aims to improve loitering detection accuracy by minimizing the ellipse area relative to A_2 and the fitting error relative to P_0. This method focus on determining the specific segment within the trajectory where loitering may occur. This is useful to detect random and abnormal trajectory types of loitering.
- **Closed Area Method:** This method identifies closed loops within the trajectory (Figure 4e) and fits ellipses to these loops. Key variables include the coordinates of the trajectory points, the geometric area of the fitted ellipses, and an area threshold A_3. The algorithm iteratively computes lines between points to find intersections, which are compared against A_3 to determine loitering. This focuses on detecting random walks and loops.

In our methodology, we focus primarily on unsupervised methods for loitering detection through trajectory analysis. However, we will also evaluate supervised approaches on top of trajectory features to provide a more comprehensive comparison thanks to the new annotated data. Specifically, we utilize Random Forest and Multi-layer Perceptron (MLP). The aim is to determine how these well established learning methods (in the presence of annotated data to train) can better recognize patterns of loitering behavior compared to the unsupervised scenario.

5. Experiments and Results

Our experimentation primarily focuses on trajectory data analysis. The primary objective is to determine each method's ability to detect loitering, taking into account their unique properties for identifying areas. Furthermore, we utilize machine models to evaluate whether trajectory patterns can be learned to better recognize loitering compared to the unsupervised scenario. The main aim of these experiments is to provide a comprehensive set of results to serve as reference for future research.

5.1. Implementation Details

In a restricted unsupervised scenario, annotated data can not be used for hyperparameter tuning. However, in our case we defined train-test splits where we also evaluate RF and MLP models. To compare all methods on the same test data and analyze the discriminative power of geometric methods, we also finetune their hyperparameters using the training data. For that, we use a random 20% of the training data as validation and we use it to finetune geometric models parameters.

In surveillance scenarios, trajectory lengths of monitored objects can vary. For the case of supervised methods requiring same input size dimensions for all samples, we address this by using trajectory padding to align all data to a standard length of 120, chosen either empirically or based on dataset features. Shorter trajectories are extended using a placeholder coordinate of (-1, -1), and single-point trajectories are duplicated, assuming stationarity, to conform to this standard size.

After trajectory padding, we compute trajectory features for the supervised methods by simply concatenating (x,y) trajectory pairs into a 1D vector. Random Forest is employed with 100 trees. Overfitting is mitigated with minimum samples split at 2 and minimum samples leaf at 1. The max depth is unlimited to allow the trees to grow and capture sufficient complexity from the data. The Multi-Layer Perceptron (MLP) consists of an input layer, followed by two hidden layers with 100 and 50 neurons, and finally an output layer. The activation function is the Rectified Linear Unit (ReLU), and optimization is performed using Stochastic Gradient Descent (SGD) with an initial learning rate of 0.0001. To avoid overfitting, a L2 regularization term is set to 0.0001. The total number of learnable parameters in this architecture is 29,201.

5.2. Evaluation metrics

We assessed our loitering detection system using the following metrics: Precision, Recall, F1-score, and ROC AUC. The F1-score, a harmonic mean of precision and recall, serves as a particularly informative metric. It favors methods that achieve a balance between precision and recall, enhancing the system's overall capability for effective loitering detection. False positives and false negatives are of critical concern. False positives not only lead to unnecessary surveillance and resource waste but can also raise ethical issues related to wrongful accusation. Conversely, false negatives can compromise security by failing to identify actual loitering events. Therefore, minimizing both types of errors is vital for the system's effective and ethical operation.

5.3. Results

The results presented in this section are organized in two categories. First, we provide quantitative results that offer a detailed evaluation of our performance metrics in both unsupervised and supervised settings. Second, we showcase visual results, illustrating trajectory classifications, to supplement our numerical findings.

5.3.1 Quantitative Results

Firstly, it is essential to note that each analytical method has distinct hyperparameters. Each method utilized in our study has its distinct area threshold, underscoring the tailored approach we adopt for each technique. In subsequent tables and discussions, we shed light on the results derived from these configurations and explore the implications of our findings in the test set.

Given the simplicity of the proposed methods, the evaluation process only requires a few seconds to assess the entire test set on a standard computer without any optimization procedure. The parameter settings for the various loitering detection methods are as follows: angle threshold (θ) of 13 degrees to denote tight turns. Specific area thresholds vary by method: 72.26 for the Ellipse method, 59.70 for the Convex Hull method, 196.76 for the Rectangle method, 14 for the Closed Area method. A larger area in the Convex Hull and Rectangle methods implies a non-linear trajectory. Conversely, the Closed Areas method uses a much smaller threshold area, effectively identifying many random walks where trajectories intertwine finding small areas. For no-motion methods, both short and long duration criteria are utilized: a standard deviation of 22.16, a radius of 60, with frame thresholds of 7 for short and 94 for long duration. The large radius is employed solely for visual representation.

Observing the unsupervised geometric methods results from Table 1, one can see that Closed Area stands out by scoring the highest in both Precision and ROC AUC. This suggests that Closed Area's region-based approach is more adept at capturing the nuances of loitering behavior, especially when compared to other shape-centric methods like Rectangle and Convex Hull. These latter methods, despite their simplicity and computational efficiency, fall short in capturing the complex spatial patterns associated with loitering, as reflected in their lower Precision and ROC AUC scores. The No Motion method, particularly its short-term

Table 1. Results for Loitering Classification with Inverse Perspective Matrix and Trajectory Smoothing pre-processing.

Method	Precision	Recall	F1-Score	ROC AUC
Rectangle	0.157	0.274	0.199	0.458
Convex hull	0.159	0.275	0.202	0.461
Ellipse	0.129	0.186	0.152	0.440
Closed Area	**0.484**	**0.395**	**0.435**	**0.646**
No Motion (Short-term)	0.064	0.123	0.084	0.344
No Motion (Long-term)	0.249	0.220	0.233	0.529
Random Forest	**0.663**	**0.471**	**0.551**	**0.706**
MLP	0.617	0.248	0.354	0.605

application, shows the lowest score across all metrics. This is indicative of the method's limited utility in this specific context.

In contrast to geometric methods, supervised machine learning techniques, show a marked improvement in performance. Random Forest, in particular, emerges as the best-performing method. The strength of Random Forest likely lies in its ensemble approach, able to model complex data distributions and being robust against overfitting. MLP also performs commendably, particularly in ROC AUC, suggesting that neural networks could offer a viable alternative for loitering detection. Though not as effective as Random Forest in our study, MLP's strong performance in ROC AUC provides a basis for future research into the optimization of neural network-based methods for this application. The main limitation of MLP in our scenario could be the limited amount of annotated data compared to the number of model parameters to be optimized 29,201 learnable parameters.

It is important to note that the methods used in this study can serve as a baseline for the first time on this specific dataset. While they are simple, they are efficient and their application to this dataset provides valuable insights on the characterization of loitering. The results reveal a clear distinction between the effectiveness of unsupervised geometric methods and supervised machine learning methods, with the latter generally outperforming. This not only emphasizes the importance of method selection but also highlights the potential of the dataset itself as a platform for more advanced research.

5.3.2 Qualitative Results

First, we showcase some false positives and false negatives in loitering detection using geometric algorithms. In Figure 5a and Figure 5b, the trajectories in confined areas are misclassified as normal behavior by the Closed Area and Convex Hull algorithms, respectively. This demonstrates some of the weaknesses of geometric methods. They do not fully utilize temporal information, and the geometric thresholds can be sensitive.

(a) (b) (c) (d)

Figure 5. Selected failed cases with the geometric algorithms. Blue lines represent trajectories, and red lines indicate the area threshold. (a) False positive: loitering detected by Closed Areas. (b) False positive: loitering detected by Convex Hull. (c) False negative: no loitering detected by any geometric algorithms. (d) False negative: no loitering detected by any geometric algorithms.

(a) (b) (c) (d)

Figure 6. Selected successful cases illustrating the effectiveness of the Closed Area and Random Forest algorithms. (a–b) Loitering detected by Closed Area but not by any other geometric methods. (c–d) Loitering detected by Random Forest but not by any of the geometric methods.

Figure 5c and Figure 5d illustrate cases where the geometric algorithms fail to detect genuine instances of loitering. These trajectories show individuals lingering in specific areas or making subtle changes in direction, indicating loitering behavior. This highlights the geometric algorithms' insensitivity to subtle shifts in movement direction. However, it's important to note that the supervised methods in our study, namely Random Forest and MLP, successfully identify these instances of loitering.

In our study, Closed Area and Random Forest prove to be the most effective unsupervised (geometric) and supervised methods, respectively. Figure 6a and Figure 6b demonstrate cases where Closed Area detects loitering, while other geometric methods do not. On the other hand, Figure 6c and Figure 6d showcase loitering cases detected by Random Forest but not by other algorithms.

The challenges associated with annotation uncertainties are noteworthy. For instance, an individual might frequently change directions within a confined space, potentially leading to interpretations of their behavior as loitering. However, such behavior could also suggest other activities, such as searching or waiting. Thanks to the openness of our dataset, future efforts could focus on creating more nuanced annotations to improve its versatility and usability.

6. Conclusions

In this work, we presented an annotated loitering dataset that maintains privacy due to the use of thermal images. Our contribution is the annotations of loitering instances in Long-term Thermal Drift Dataset with a loitering protocol definition as, seated, no motion, abnormal trajectory and random trajectories, with in total of 19,737 trajectories annotated. This dataset is the largest of its kind in terms of loitering annotations, providing a solid base for further research. We evaluated the dataset using both unsupervised trajectory analysis and supervised methods like Random Forest (RF) and Multi-layer Perceptron (MLP). While RF shows higher performance, it relies on labeled data. On the other hand, geometric descriptors, particularly closed areas, also yield high performance. Importantly, these descriptors are more adaptable across different datasets without the need for labeled data. MLP, while effective, demands a large volume of labeled data for optimal performance.

The subjectivity in trajectory behaviors influences the binary annotation for loitering, necessitating a precise definition to mitigate interpretation biases. Future work could define loitering's start and end boundaries within a trajectory, introduce multiple loitering categories, and annotate local points of interest like ATMs or shops for contextual analysis. Considering the limitations of supervised methods like RF and MLP for labeled data, exploring the adaptability of geometric descriptors across datasets is promising. It's crucial to recognize potential demographic biases since loitering definitions can vary culturally and subjectively. Therefore, ethical algorithm development in this field requires a multidisciplinary approach to maintain integrity.

Acknowledgements. This work has been partially supported by the Spanish project PID2022-136436NB-I00 and by ICREA under the ICREA Academia programme.

References

[1] Pets2007: Dataset for surveillance. page http://www.cvg.reading.ac.uk/pets2007/data.html, 2007. 3

[2] Amit Adam, Ehud Rivlin, Ilan Shimshoni, and Daviv Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(3):555–560, 2008. 3

[3] Xinyang Feng, Dongjin Song, Yuncong Chen, Zhengzhang Chen, Jingchao Ni, and Haifeng Chen. Convolutional transformer based dual discriminator generative adversarial networks for video anomaly detection. *Proceedings of the 29th ACM International Conference on Multimedia*, 2021. 2

[4] Yepeng Guan and Wenqing Mao. Pedestrian virtual space based abnormal behavior detection. *IAENG International Journal of Computer Science*, 46(2), 2019. 1, 2

[5] Liu Jianquan and Nishimura Shoji. Automatic classification of behavior patterns for high-precision detection of suspicious individuals in video images. 2019. 2

[6] Dohun Kim, Heegwang Kim, Yeongheon Mok, and Joonki Paik. Real-time surveillance system for analyzing abnormal behavior of pedestrians. *Applied Sciences*, 2021. 1, 2

[7] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 4

[8] Wenting Li, Dongping Zhang, Min Sun, Yibo Yin, and Ye Shen. Loitering detection based on trajectory analysis. In *2015 8th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, pages 530–533, 2015. 1, 2

[9] Wenting Li, Dongping Zhang, Min Sun, Yibo Yin, and Ye Shen. Loitering detection based on trajectory analysis. In *2015 8th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, pages 530–533. IEEE, 2015. 2

[10] Wei-Xin Li, Vijay Mahadevan, and Nuno Vasconcelos. Anomaly detection and localization in crowded scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 36:18–32, 06 2013. 3

[11] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 fps in matlab. In *2013 IEEE International Conference on Computer Vision*, pages 2720–2727, 2013. 3

[12] Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked rnn framework. In *Proceedings of the IEEE international conference on computer vision*, pages 341–349, 2017. 3

[13] Shanmughapriya M, Gunasundari S, and S. Bharathy. Loitering detection in home surveillance system. *2022 10th International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-22)*, pages 1–6, 2022. 2

[14] Ramin Mehran, Alexis Oyama, and Mubarak Shah. Abnormal crowd behavior detection using social force model. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 935–942, 2009. 3

[15] Rashmiranjan Nayak, Mohini Mohan Behera, Vaisyaraju Girish, Umesh Chandra Pati, and Santosh Kumar Das. Deep learning based loitering detection system using multi-camera video surveillance network. *2019 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS)*, pages 215–220, 2019. 2

[16] Rashmiranjan Nayak, Mohini Mohan Behera, Vaisyaraju Girish, Umesh Chandra Pati, and Santos Kumar Das. Deep learning based loitering detection system using multi-camera video surveillance network. In *2019 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS)*, pages 215–220, 2019. 2

[17] Ivan Adriyanov Nikolov, Mark Philip Philipsen, Jinsong Liu, Jacob Velling Dueholm, Anders S. Johansen, Kamal Nasrollahi, and Thomas Baltzer Moeslund. Seasons in drift: A long term thermal imaging dataset for studying concept drift. In *NeurIPS Datasets and Benchmarks*, 2021. 1, 3

[18] Hiroyuki Ootomo and Rio Yokota. Recovering single precision accuracy from tensor cores while surpassing the fp32 theoretical peak performance. *The International Journal of High Performance Computing Applications*, 36:475 – 491, 2022. 2

[19] Luis Patino and James Ferryman. Pets 2014: Dataset and challenge. In *2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 355–360, 2014. 3

[20] Bharathkumar Ramachandra and Michael Jones. Street scene: A new dataset and evaluation protocol for video anomaly detection. *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2558–2567, 2019. 3

[21] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *ArXiv*, abs/1804.02767, 2018. 2

[22] Arivazhagan Selvaraj and Newlin Shebiah Russel. Versatile loitering detection based on non-verbal cues using dense trajectory descriptors. *Multimedia Tools and Applications*, 78:10933–10963, 2019. 2

[23] Wonsup Shin, Seok-Jun Bu, and Sung-Bae Cho. 3d-convolutional neural network with generative adversarial network and autoencoder for robust anomaly detection in video surveillance. *International journal of neural systems*, page 2050034, 2020. 2

[24] Muhammed Shuaau, Ka Fei Thang, and Nai Shyan Lai. Autonomous abnormal behaviour detection using trajectory analysis. *International Journal of Electrical and Computer Engineering (IJECE)*, 2019. 2

[25] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649. IEEE, 2017. 2

[26] Hongchun Yuan, Zhenyu Cai, Hui Zhou, Yue Wang, and Xiangzhi Chen. Transanomaly: Video anomaly detection using video vision transformer. *IEEE Access*, 9:123977–123986, 2021. 2

[27] Zhihao Zhang, Liang Huang, Xin Peng, Yuanqiao Wen, and Lifei Song. Loitering behavior detection and classification of vessel movements based on trajectory shape and convolutional neural networks. *Ocean Engineering*, 2022. 2