

Privacy Preserving Solution for People Flow Origin-Destination Matrix Estimation in Public Areas

Enrico Collini, Marco Fanfani, Luciano Alessandro Ipsaro Palesi, Matteo Marulli, and Paolo Nesi(✉)

Department of Information Engineering, Distributed Systems and Internet Technology Lab (DISIT), University of Florence, Via Santa Marta, 3, 50139 Florence, Italy
{enrico.collini,marco.fanfani, lucianoalessandro.ipsaropalesi}@unifi.it, {matteo.marulli, paolo.nesi}@unfi.it
http://www.disit.org/, http://www.snap4city.org/

Abstract. Assessing how people typically move across public areas like city squares, shopping centers, airports, train stations, metro hubs, etc., is a relevant task required to both perform informed decisions for infrastructure design and to identify unusual behaviors to support surveillance activities. Thanks to the development of computer vision algorithms, methods for detecting and tracking pedestrians offer nowadays valid solutions to tackle this problem. However, most of the state-of-the-art approaches exploit color images that can pose threats to people's privacy. Additionally, the classification of trajectories to estimate local origin-destination matrices and enable analysis of typical people flow is still poorly addressed. Therefore, in this paper, we propose a complete solution able to detect, track, and classify people flows in complex junctions by exploiting privacy-preserving thermal cameras. Thanks to a thorough examination of several machine learning methods, and to the identification of a new set of features for trajectory classification, also evaluated with eXplainable AI approaches, the proposed solution has been capable to obtain accurate trajectory classification – achieving an F1-score of 0.96 – that can be exploited to compute local origin-destination matrices. Moreover, early classification in real time using partial trajectories is possible. The solution has been implemented exploiting the Snap4City Smart City Digital Twin infrastructure and has been carried out in the context of CN MOST, the Italian National Center on Sustainable Mobility.

Keywords: People Flows · Origin-Destination Matrix · Tracking · Trajectory Classification · Early Classification · Thermal Imaging

1 Introduction

Nowadays there is a relevant interest from municipalities and private companies to understand how people move within public locations like squares, shopping centers, retails, airports, metro and multimodal hubs, supermarkets to address security, planning,

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2026
O. Gervasi et al. (Eds.): ICCSA 2025 Workshops, LNCS 15895, pp. 111–128, 2026.
https://doi.org/10.1007/978-3-031-97651-3_8

and/or commercial purposes. People flow data can be utilized to compute local Origin-Destination Matrices (ODMs) by extracting typical paths/trajectories, so as to identify suspicious behaviors and understand user behaviors for multiple purposes. Obtaining such insights can be fundamental in operational management scenarios where real-time analytics are required for monitoring and producing early warnings: for example, a rapid detection of flow changes in a metro hub could be due to some critical occurring events. On the other hand, when it comes to decision makers, a more general analysis of trajectories and flow distribution over time could be very useful to study people's movements and adapt the services offered.

Color RGB cameras are widely used for this purpose. However, with the adoption of the European General Data Protection Regulation (GDPR) [1], a relevant number of restrictions has made more difficult to obtain the authorizations needed to deploy RGB devices, even when security concerns are invoked. The European Commission's deliberations attest a broader global discourse on the balance among security, public safety, and protection of individual privacy rights [2]. To address these limitations, the usage of more privacy-preserving sensors has started, like laser [3] and anonymized Wi-Fi/Bluetooth sniffing [4] sensors. Despite these efforts, such solutions are mainly unsuitable to understand flow distributions at junctions and in general human behavior. More interesting is thermal imaging that can reduce issues related to privacy by capturing only heat signatures of the elements and that can be exploited with advanced computer vision algorithms.

In the literature, detection and tracking of people have been addressed by computer vision, mostly exploiting RGB images [5]. More recently, convolutional neural networks have been adopted [6–9]. In [10], the authors proposed an efficient indoor people detection method based on RGB-D cameras. The method extracts RGB and depth features and fuses them to get richer feature-maps improving detector performance. Yan et al., [11] proposed a solution for detecting and tracking people in indoor environments using a robot equipped with an RGB-D camera to detect the underside of a person using Histogram Oriented Gradients (HOG) followed by a pre-trained Soft Support Vector Machine (soft-SVM) binary. Then, tracking was achieved using a Bayesian filter with a constant velocity model. However, such approaches may not work well in real scenarios with crowds. A solution for a multi-target multi-camera task for vehicles has been proposed in [12]. The system, based on YOLOv7 detector and DeepSORT tracker [13] achieved good performance with some limits due to lighting, weather conditions, and vehicle dimensions. More advanced methods for people detection in crowded scenarios have also been proposed. In [14], the problem was addressed with a tracking-by-detection method using Mask R-CNN detector [15], ReID-Net for target association, and LSTM-based network [16] to deal with occlusions and motion blur. Benahmed et al., [17], used a deep learning approach for detecting and tracking honeybees, using YOLOv5 as object detector and StrongSORT [18] for tracking. You, Li, and Zhang [19] used RGB-D sensors for object detection with YOLOv5 and tracking with DeepSort exploiting a bird's-eye camera view, to help YOLOv5 to distinguish better individuals and minimizing occlusion for DeepSort, thus preventing ID switches at the expense of limiting the possibility of applying the solution in actual scenarios. Note that, since all these methods employ RGB cameras people's identities may be revealed and privacy violated.

Regarding the usage of thermal cameras, some works on people detection were carried out, by using HOG-like features [20], saliency map generation [21], or YOLO models trained on thermal images [22]. Differently, less attention was devoted to tracking solutions exploiting thermal data. For example, in [23] thermal cameras have been used to achieve detection and tracking simultaneously. However, the tracking task has been limited to only one person at a time.

1.1 Paper's Aim and Organization

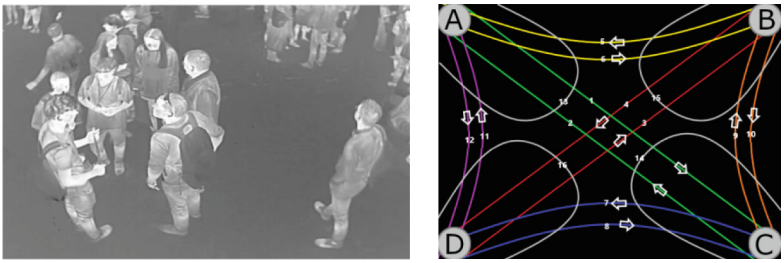
As above described, current state-of-the-art works have focused on detection and tracking systems, mainly based on RGB images. The solution and results described in this paper have been focused on exploiting privacy-preserving thermal images, intrinsically GDPR compliant, to detect, track, and identify people's trajectories to understand in real time their behaviors in crowded junctions. Therefore, this paper's main contributions are:

1. Assessing the performance of state-of-the-art solutions for people detection and tracking, when thermal cameras are used instead of RGB sensors;
2. Developing a novel thermal camera-based solution to recognize, classify, and thus count people's trajectories, exploiting novel features validated with eXplainable AI (XAI) approaches, and obtaining local origin-destination matrices for complex junctions;
3. Extending the proposed solution to offer early classification of trajectories, which is enabled by real-time detection and tracking, and classification on the basis of partial trajectories. The early classification of a trajectory without waiting for its completion can be very useful to detect anomalies and enable early warnings.

The solution proposed in this paper includes detection, tracking, and trajectory classification, working in real-time and simultaneously on multiple targets. Innovative aspects, not covered by the state-of-the-art solutions, are related to real-time exploitation of thermal cameras for the above-described points (2) and (3). The computing of multiple people's trajectories and their recognition, classification and counting in real time by means of thermal data is not a trivial task, due to the limited information contained in thermal images, and has not been widely addressed in literature. Furthermore, the proposed solution is characterized by the implementation of direction-based trajectory classification, achieved by a thorough process of feature engineering and data processing. Using XAI techniques, the most relevant features for the classification of trajectories have been identified and used. Moreover, the use of XAI emphasizes both transparency and comprehensibility of the model, which are crucial aspects in any deep learning-based solution. Indeed, this approach contributes to the accuracy of trajectory classification and provides a clear interpretation of model decisions through the identification of key features involved in the process.

The proposed solution has been tested and validated using video streams taken from thermal cameras in crowded junctions (with an average of more than 26 persons per frame) captured during a major international expo in Barcelona, and in Florence. The solution has been implemented by exploiting the Snap4City Smart City Digital Twin framework and infrastructure (www.snap4city.org) [24–26] which is an open-source

FIWARE platform. This research was carried out in the context of CN MOST, the Italian National Center on Sustainable Mobility of the Italian Ministry of research [27].



(a) (b)

Fig. 1. (a) Example of image taken from thermal camera in a crossing area. (b) Corresponding possible paths of the same crossing area.

The paper is organized as follows. In Sect. 2, the application context is presented. Section 3 provides an overview concerning the main modules of this proposed approach. In Sect. 4, experimental results are reported for detection, tracking, and classification. In Sect. 5, conclusions are drawn.

2 Context Description

The work described in this paper has focused on developing a solution based on thermal cameras for detecting, tracking, and classifying multiple people flows in a hub junction, and thus obtaining a resulting local ODM.

In Fig. 1a, the thermal image of a junction is presented, while Fig. 1b describes possible trajectories in the same junction reporting as A, B, C, D the four entering/exiting locations. The junction in the example is an X junction where the people may arrive from any direction to go in any other direction, with possible stops in the middle. A total of 16 different paths are possible (the numbered IDs are reported in Fig. 1b) including the ones where people enter and exit from the same direction. This is what happens in a real-case scenario, such as at an expo, in train stations, metro hubs, shopping centers, etc. It should be remarked that paths shown in Fig. 1b are a categorization of real trajectories that typically are noisier and more confused.

To achieve such an objective, it is required to detect multiple pedestrians and track them simultaneously, dealing with partial occlusions and irregular movements (e.g., stop-and-go), and avoiding identity switches (i.e., mixing trajectories due to the overlap of two or more trajectories). Finally, the estimated trajectories have to be classified into defined paths. In this paper, we carried out an in-depth analysis of state-of-the-art solutions for detecting and tracking people using thermal cameras as the only input. Then, we devised a set of features and compared performances of several classifiers trained from scratch with specific implementations to associate each trajectory to one of the 16 possible paths, in real-time, even in an early classification modality exploiting partial trajectories. Note that, even if some studies on pedestrian detection with thermal

cameras have been carried out, as reported in the previous section, no trained models are freely available.

The use case, under analysis in the experiments and for which the solution has been validated, is related to an X junction which is the most complex real case junction.

3 Data Flow and Method Structure

As presented in the Introduction, the objective is to develop a solution that exploits thermal cameras to detect and track pedestrians, as well as to classify their trajectories and obtain local ODMs. Hereafter the phases of the solution are described.

Phase (A) – Thermal Image Acquisition. A thermal camera provides a video stream, out of which single frames are extracted. The thermal camera used for data acquisition is a wide-angle AXIS Q1952-E. The radial distortion in the captured frame due to the wide-angle lens was corrected with appropriate functions [28] before any processing of images in Phase (B). Several kinds of cameras can be used according to the space; therefore, correction is needed to process images with common models.

Phase (B) - People Detection. This module detects multiple pedestrians in camera frames and extracts their positions and bounding boxes by using an object detection process. The usage of both YOLOv5 [29] and YOLOv8 [30] has been evaluated and compared assessing their performance on thermal imaging, following the work presented in [31, 32]. The best performing method working on thermal cameras has been selected for the proposed solution.

Phase (C) - People Tacking. This phase considers several tracked positions to create trajectories. To this end, algorithms like DeepSORT [13] fail to perform continuous tracking when tracked objects are occluded or when they perform chaotic movements as described above, since they are based on Kalman Filter [33] and do not use any strategy neither to recover lost tracks, nor to limit accumulation of error in the tracked object states. In this work, more recent tracking solutions have been evaluated, including byteTrack [6], OC-Sort [34], and StrongSort [18], which exploit advanced mechanisms to address the issue of lost tracking. Comparative results are reported in Sect. 4. Note that, when a person exits the junction and re-enters it at a successive time, two different trajectories are instantiated. This is not a problem since the objective is to create the ODM at junctions. When re-identification is needed, alternative methods could be used [35]. Thanks to the tracking phase, it is possible to obtain a trajectory for each person. However, the resulting trajectories from Phase (C) are very noisy to be classified in a reliable way.

Phase (D) – Trajectory Filtering/Smoothing. This step is devoted to smoothing any estimated trajectories before classification. To this end, Gaussian filtering has been adopted. Given a trajectory described by a sequence of 2D points (x, y) expressed in the canonical image reference system (with x-axis from left to right, and y-axis from top to bottom), two independent Gaussian filters are used on x and y coordinates, with a common standard deviation σ that depends on the trajectory length (i.e., the number of

samples) according to the following equation:

$$g(x, \sigma_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{1}{2}\left(\frac{x}{\sigma_i}\right)^2\right),$$

$$\sigma_i = 0.1 \times T_i,$$

where: T_i is the length of the i-th trajectory.

Phase (E) – Feature Extractions from Trajectories. To prepare data for trajectory classification a set of features has to be identified and computed. Trajectories may have several basic features such as: starting and ending points, start and end timestamps, a vector of points, person bounding boxes over time with their sizes, etc. Derived features can be duration in time, size of the trajectory in terms of points or meters or pixels, etc. Features have to be extracted from the resulting data of **Phase (D).** Please note that each trajectory is a polyline, and thus it provides a variable number of points over time. However, the successive phase of recognition/ classification requires a fixed size input vector of features to work.

Phase (F) – Recognition and Classification of Trajectories. In this phase, each trajectory is classified in one of the 16 possible paths (see Fig. 1b). Recognition and classification can be performed using several different techniques which can be classified as supervised and non-supervised. In the context of the proposed solution, several models have been tested and compared, and in particular: LR (Logistic Regression) [36], XGBoost [37], KNN (k-nearest neighbors) [38], SVM (Support Vector Machine) [39], K-Means clustering [40]. Comparative results are reported in Sect. 4.

4 Experimental Analysis and Results

In this section, the evaluation of the proposed solution for pedestrian detection and tracking based on thermal images and the obtained results on trajectory classification are reported. Different metrics have been used to assess the performance of the algorithms for the tasks of detection, tracking and path classification: they are reported hereafter, together with a description of the used datasets. The results achieved are shown and discussed in the following sections. As stated before, innovation relies on the use of thermal cameras, the definition of trajectory features considering XAI approaches, the possibility of an early classification (exploiting partially traveled trajectories) and the real time computation.

4.1 Datasets and Metrics

Experiments were evaluated using a dataset of thermal videos recorded with an AXIS Q1952-E camera, referred to as AQ1952 dataset. In Fig. 1a, an example frame is reported. The data were recorded during an expo conference, when capturing a crowed scene at an X junction. A total of 16 possible paths is considered (as represented in Fig. 1b).

AQ1952 included video sequences covering a time span of about an hour during which 1696 trajectories were identified, with an average of 26.97 people for each frame.

Video frames have been preprocessed to remove radial distortion due to the wide-angle lens of the camera, since such kind of distortion is usually not considered in pre-trained detectors. Moreover, without distortion, the training set and models resulted in being more re-usable since different cameras may have different lens and distortions. Ground-truths for the detection, tracking, and classification tasks have been manually produced. Each video frame has been manually labeled with bounding boxes describing the pedestrian positions. Then, sequences of bounding boxes of the same person were created by associating the detections in subsequent frames to obtain ground-truth tracks. Finally, the resulting trajectories were manually labeled with one of the 16 paths.

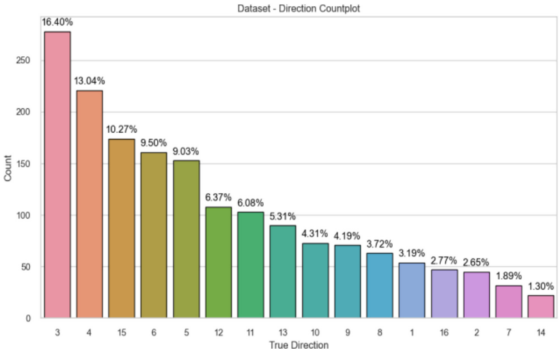


Fig. 2. Distribution of trajectory classes in AQ1952.

As shown in the histogram of Fig. 2, the dataset is unbalanced in terms of trajectory classification, with classes 3 and 4 being the most common. A re-balancing has been carried out for the training phase by limiting the number of paths in classes 3 and 4 at max 200 examples, so as to have a ratio of 10:1 between the largest and the smallest class. Differently, we kept the original class distribution for the testing phase to get a more realistic assessment.

Additionally, the LLVIP dataset [41] was also used to fine-tune the pedestrian detectors to work on thermal data. Note that LLVIP is grounded with images without distortions, one reason more to remove distortions on the AQ1952 dataset.

To quantify the performance of the different modules/phases the following metrics were used. People detection is evaluated using Precision (P), Recall (R), and the mean average precision (mAP) [19] with specific Intersection over Union (IoU) thresholds to count the number of True Positive (TP), False Positive (FP), and False Negative (FN). For people tracking, the Higher Order Tracking Accuracy ($HOTA$) [42] was selected, since it combines measures of localization accuracy and identity switches to assess tracking quality incorporating both spatial accuracy and temporal consistency. $HOTA$ score is computed according to the following equations:

$$HOTA = \sqrt{\frac{\sum_{c \in \{TP\}} A(c)}{|TP| + |FN| + |FP|}}$$

$$A(c) = \frac{|TPA(c)|}{|TPA(c)| + |FNA(c)| + |FPA(c)|}$$

where *TPA*, *FNA*, and *FPA* are the True Positive Association, False Negative Association, and False Positive Association respectively, for a given *TP* detection c. . Finally, to assess the trajectory classification, Precision (P), Recall (R), and F1-score ($F1$) were used [43]. In this case, since the test dataset remained unbalanced, weighted versions (wP, wR, $wF1$, respectively) have been preferred, and computed according to the following formula:

$$\frac{1}{\sum_{c \in C} |\hat{y}_c|} \sum_{c \in C} |\hat{y}_c| \phi(y_c, \hat{y}_c)$$

where C is the number of classes, $\hat{y}_c$ are the true class labels and y_c the predicted ones, $|\hat{y}_c|$ indicates the true number of examples in class c, and $\phi(y_c, \hat{y}_c)$ computes Precision, Recall, or F1-score as usual.

Table 1. Results for object detection Phase (B).

Model	Precision (↑)	Recall (↑)	mAP@50 (↑)	mAP@[50–95] (↑)
YOLOv5 [29]	**0.911**	**0.815**	**0.930**	**0.534**
YOLOv8 [30]	0.911	0.788	0.895	0.498

4.2 Evaluation of People Detection and Tracking

For people detection, carried out in Phase (B) of the proposed solution, pretrained versions of YOLOv5 and YOLOv8 were finetuned on thermal images using the LLVIP dataset, augmented as in [32] to improve its generalization to different people dimensions in images, and the AQ1952 dataset produced. Tests were conducted on the test set of AQ1952. As shown in Table 1, YOLOv5 achieved the best results in terms of mAP@50 (i.e., the IoU threshold was set to 0.50) and other metrics.

For Phase (C), devoted to pedestrian tracking, three different state-of-the-art approaches have been evaluated: byteTrack [6], OC-Sort [34], and strongSort [44], trained and tested on AQ1952. The above-mentioned *HOTA* metric has been used to assess the capability of the evaluated approaches to associate the detection in the current frame with the same person from the previous frame. Note that this assessment does not take into account the noise level of the trajectory and neither if the trajectory can be profitably and correctly classified by classification module of Phase (E). Results are reported in Table 2. OC-Sort obtained the best score in terms of *HOTA* and thus it has been selected for the proposed framework.

The trajectories obtained are smoothed by Gaussian filtering to reduce the noise estimated in Phase (D). Then, the resulting trajectories are used as input for the feature extraction module of Phase (E).

Table 2. Results for object tracking Phase (C).

Algorithm	HOTA (↑)
byteTrack [6]	50.673
OC-Sort [34]	**56.772**
strongSort [44]	46.950

Table 3. List of features used for trajectory description.

Feature	Description
trajectory	list of (x, y) trajectory coordinates in the image coordinate reference system
npoints	number of (x, y) points in the trajectory
s_x	x-value of the start point of trajectory
s_y	y-value of the start point of trajectory
e_x	x-value of the final point of trajectory
e_y	y-value of the final point of trajectory
dx	$e_x - s_x$ difference
dy	$e_y - s_y$ difference
m	the slope of the line connecting the starting and ending point of trajectory
q	*y*-intercept of the line connecting the starting and ending point of trajectory
angle	angle of the trajectory. The angle is calculated using the arctangent of the slope of the line between the initial point of the trajectory and the final point
curvature	curvature of the trajectory. To calculate curvature, three consecutive points along the trajectory were selected. The angles between the segments formed by these points were calculated. The process is repeated for all the three subsequent points, and finally the average of the angles is obtained
concavity	concavity of the trajectory. See the text for details
len-trajectory	the length of the trajectory calculated as the sum of the Euclidean distances of each consecutive couple of points
DistMinBarRet	the distance between the centroid of the trajectory and the line connecting a start and end points of the trajectory

4.3 Feature Extraction

In this Section, the features identified to be used in the proposed trajectory classification method are presented. The main goal was to pass from an arbitrary number of points to a set of features describing the trajectory with a fixed size vector. Initially, a larger set of features have been identified and tested. Those reported and discussed in the paper have been demonstrated to be the most relevant for the classification by using a XAI approach as explained in the following.

Table 4. Results for trajectory classification, Phase (F).

Classifier	Weighted Precision (↑)	Weighted Recall (↑)	Weighted F1-score (↑)
XGBoost [37]	**0,9620**	**0,9607**	**0,9603**
SVM [39]	0,9383	0,9371	0,9366
KNN [38]	0,8699	0,8703	0,8662
LR [36]	0,9524	0,9489	0,9491
k-means [40]	0,9205	0,9174	0,9170

In Table 3, the list of features is reported with their respective description. The first group of features includes direct characteristics, such as the list of trajectory samples (*trajectory*), the number of polyline points which are derived from the frames (*npoints*), the coordinates of the starting (s_x, s_y) and ending (e_x, e_y) position of the trajectory. Then, a series of derived features has been newly defined and computed: the signed difference along the x and y axes between the ending and starting samples (dx and dy); the slope (m) and y-intercept (q) of the line l_T passing thought the starting and ending samples; the angle of the trajectory computed as the arctangent of the slope m of the line l_T; the trajectory curvature calculated as the average of the angles between the segments obtained considering three consecutive points along the trajectory. The concavity is determined by evaluating the following equation:

$$Concavity = \begin{cases} +1, \textit{ if } A > 0 \\ -1, \textit{ if } A < 0 \\ 0, \textit{ otherwise} \end{cases}$$

where $A = s_x m_y + s_y e_x + m_x e_y - e_x m_y - s_x e_y - m_x s_y$, and (m_x, m_y) is the middle trajectory sample. Note that A is derived from the formula for the signed triangle area, i.e., the area of a triangle given by (s_x, s_y), (m_x, m_y), (e_x, e_y), is positive if the vertices are listed counterclockwise, or negative otherwise. The trajectory length is represented as the sum of Euclidean distance among consecutive samples. Finally, *DistMinBarRet* is the Euclidean distance between the middle trajectory sample and line l_T passing through the starting and ending samples.

4.4 Trajectory Classification

The final Phase (F) of the proposed approach performs the trajectory classification according to the 16 classes of paths defined in Sect. 2 (see Fig. 1b), thus exploiting the direct and derived features presented in the previous Section (their relevance is discussed later). Table 4 reports the comparative classification results considering the following techniques: LR, XGBoost, KNN, SVM, and k-means. For all of them, an optimization for hyper-parametrization has been conducted using cross-validation and the Bayesian search of Optuna [45]. As shown, the most promising solution turned out to be XGBoost, which is based on decision trees able to handle strongly nonlinear feature space. XGBoost achieved a wP of 0.9620, a wR of 0.9607, and a $wF1$ of 0.9603, thus

Table 5. Classification report on the test set for each class using XGBoost, Phase (F).

Path	Precision (↑)	Recall (↑)	F1-score (↑)
1	1.0000	0.9375	0.9677
2	1.0000	0.8571	0.9231
3	0.9655	1.0000	0.9825
4	0.9565	1.0000	0.9778
5	1.0000	0.9130	0.9545
6	0.9574	0.9375	0.9474
7	0.8182	0.9000	0.8571
8	0.9500	1.0000	0.9744
9	0.9130	1.0000	0.9545
10	1.0000	1.0000	1.0000
11	0.9630	0.8387	0.8966
12	0.9677	0.9375	0.9524
13	0.9630	0.9630	0.9630
14	1.0000	0.8571	0.9231
15	0.9455	1.0000	0.9720
16	0.9333	1.0000	0.9603

outperforming all the other methods. Tables 5 shows the results related to XGBoost classification for each of the different trajectory classes that were considered. Note that the proposed implementation of XGBoost has considered the generalized form of log-loss with elastic-net regularization. The multi-log loss $\mathcal{L}_{MULTI}$ is defined as

$$\mathcal{L}_{MULTI} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(p_{ij}) + \lambda_1 \sum_{w \in W} |w| + \lambda_2 \sum_{w \in W} w^2$$

where: N is the total number of observations, C is the total number of classes, y_{ij} is a binary variable indicating whether observation i belongs to class j (1 if true, 0 otherwise), p_{ij} is the predicted probability that observation i belongs to class j, λ_1 and λ_2 are hyperparameters for L1 and L2 regularizations and w are the model's weights. For completeness, in Table 6, simple and weighted average XGBoost classification scores for all the path classes are reported, while in Table 7, the XGBoost hyper-parametrization ranges and best model values are reported. According to Optuna, the most relevant parameters observed have been *subsample* (40%), η (27%), *n_estimators* (12%), and α (11%). The best results were obtained with 120 training epochs.

In order to further assess and explain the XGBoost model classification in terms of feature relevance, the XAI approach of SHAP has been exploited [46]. As can be observed from the plot reported in Fig. 3, the most relevant variables turned out to be: the starting and ending point components, arrival and departure delta in terms of distance

Table 6. Results Average results on the test set for XGBoost, Phase (F).

AVG	Precision (↑)	Recall (↑)	F1-score (↑)
simple	0.9583	0.9463	0.9507
weighted	0.9620	0.9607	0.9603

Table 7. Optimized parameters for XGBoost, Phase (F).

Parameter	Range	Description	Value
λ	$[4 * 10^{-4}, 1]$	parameter for L2-regularization	0.24953
α	$[4 * 10^{-4}, 1]$	parameter for L1-regularization	0.01450
n_estimators	[100,1000]	the number of trees	812
max_depth	[2,100]	depth of a tree	78
η	$[4 * 10^{-4}, 1]$	learning rate	0.06784
subsample	[0,1]	fraction of examples used during the training for each tree of the forest	0.46912
min_child_weight	$[4 * 10^{-4}, 1]$	minimum sum of instance weight (hessian) needed in a child	0.00287
early_stopping	- - -	if there are no more improvements in validation loss, training will terminate early	10

on X and Y, and the Euclidean distance between the origin and destination. Using the variance in both directions (*dx*, *dy*), the model can easily classify trajectories belonging to paths 1, 2, 3, and 4. However, for directions 5, 6, 7, 8, 9, 19, 11, 12, the model can only rely on the variance in one or two directions, hence requiring observation of the arrival point to produce a reliable classification. The distance between the start and end points of trajectories mainly determines directions 13,14,15, and 16. These directions are loops and, therefore, borderline variations could occur in both directions. However, due to their loop nature, the distance between start and end points is smaller than for non-loop trajectories, and in order to classify loops correctly the model must also look at the components of both points. It should be noted that features such as *angle* and *concavity* turned out to be not relevant for the classification since the extracted aspects are represented by other features, such as *curvature*, *slop*, etc.

Once the trajectories have been classified, it is possible to count the number of people passing through the junction and estimate a local ODM to capture how people are likely to move in the area. After naming A, B, C, and D the four entering/exiting points of the junction (see Fig. 1b), the related ODMs (ground truth and predicted), based on the 509 trajectories used in the testing phase, are shown in Fig. 4. As can be seen, ground truth and predicted ODMs are substantially identical and indicate that the most frequent

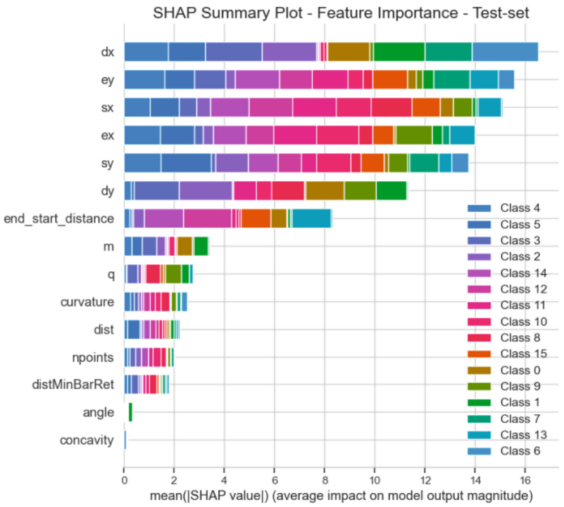


Fig. 3. Relevance of features of the XGBoost trajectory classification by using XAI SHAP, the order of the classes in the bars (from left to right) are those reported in the color legenda on the right side for top to bottom. The results are in terms of Snap values.

Origins \ Destinations	A	B	C	D
A	27	48	16	32
B	46	52	22	66
C	14	21	7	10
D	31	84	19	14

(a)

Origins \ Destinations	A	B	C	D
A	27	47	15	31
B	42	55	22	69
C	12	23	6	11
D	27	87	20	15

(b)

Fig. 4. Origin-Destination matrices (ODM) obtained from the classification of the trajectories in one of the 16 paths shown in Fig. 1b. In (a) the ground truth ODM, in (b) the predicted one.

trajectories are transversal paths from D to B and from B to D. On the other hand, C appears to be the least used origin/destination. This piece of information can be used to get insights into how people are likely to move, and, in the context of an expo (as it is for the context our analyzed data come from) it can be of help in understanding which are the most interesting areas for the general public. In different contexts, such as mobility hubs, public squares, etc., people flow and related ODMs can help to control better those areas and to plan possible improvements, to facilitate people's movements.

4.5 Early Classification of Trajectories

Early classification of trajectories is useful to enable quick anomaly detection and early warnings without waiting for the completion of each single trajectory. This goal can be obtained by performing the computation of all the Phases of the presented solution in real time and performing the classification using only partial trajectory data while the pedestrian is still moving into the scene.

A range of testing datasets with trajectories of different lengths has been generated to assess early classification using the XGBoost model and two different K-Means approaches. Trajectory lengths have been defined in relation to the diagonal of the image (considered as 100%). Different test sets have been created with trajectories of 25%, 50%, 75%, 100% and 125% length. A value of 0.6 $wF1$ has been adopted as a threshold to claim a successful early classification. Results are reported in Fig. 5, where the proposed XGBoost model was confirmed to be the best performing method in classifying trajectories also in early classification modality, meaning when the trajectories present at least 75% of their diagonal length.

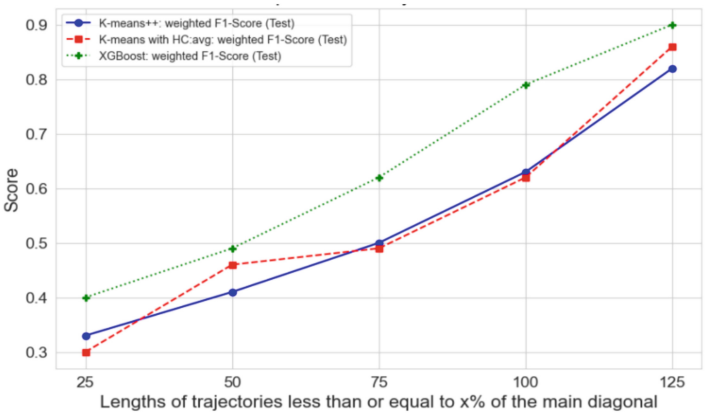


Fig. 5. Model comparison for the early classification of trajectories, in terms of F1-Score.

Table 8. Average computational times (in seconds) of the proposed solution.

Phases	Tasks	Main process	Mean (s)	std
A, B	Grab, undistort, detection	YOLOV5s	0.0855	0.0267
C, D	Tracking, smoothing	OC-SORT	0.0069	0.0019
E, F	Feature extraction, classification	XGBoost	0.0013	0.0007
All			0.0937	0.0293

4.6 Computational Times

This section shows the computational times of the proposed solution. The solution in place is capable to: grab the frames from a video stream transmitted with the RTSP protocol and eliminate radial distortion (Phase A); perform object detection, using Yolo 5 (Phase B); perform people tracking via OC-Sort (Phase C); smooth trajectories (Phase D); compute trajectory features (Phase E); and perform the related classification in the defined paths using XGBoost (Phase F). Executions have been carried out on a small workstation equipped with an Intel Core i9-10900 CPU @ 2.80 GHz, with 32 GB RAM, and an NVIDIA T1000 with 4 GB GDDR6 GPU.

Table 8 shows the average computational times required for each phase. These results demonstrate the capability of performing real-time processing: an average of 0.0937s per frame was achieved (equal to more than 10 fps), with the most time-consuming steps being acquisition and detection. Note that, usually, a thermal camera has a frame rate of about 7 fps, therefore the proposed solution is more than adequate to work in real-time on thermal video streams and low costs solutions.

5 Conclusions

This paper has proposed a solution to estimate and classify people's movements in paths describing all the possible trajectories in crowded junctions. The solution exploits state-of-the-art techniques for pedestrian detection and tracking, while using privacy-preserving thermal videos. A set of trajectory features was devised in order to train a classifier capable of assigning each estimated trajectory to a path. After a thorough examination, the YOLOv5 detector and the OC-Sort tracker were selected to implement the first steps of the pipeline. Then, using the devised features and the proposed XGBoost implementation, the system is able to classify correctly any trajectories. Moreover, by means of XAI techniques, the most relevant trajectory features have been identified, which has reduced the computational costs. Experimental results show the validity of the proposed approach, achieving very good classification performance (with a weighted F1-score of 0.9603). Experiments have also demonstrated that the proposed solution can perform early classification of trajectories starting from 75% of their length. The whole system can run in real-time on modest hardware and it is therefore easily deployable in real scenarios. The solution has been implemented by exploiting the Snap4City framework and infrastructure which is an open-source FIWARE platform. This research has been carried out in the context of CN MOST, the Italian National Center on Sustainable Mobility of the Italian Ministry of Research.

Acknowledgements. The authors would like to express sincere thanks to the MIUR, CN MOST. Special thanks go also to the many developers working on the Snap4City platforms. Snap4City (https://www.snap4city.org) and Km4City are open technologies of the DISIT Lab.

References

1. EU General Data Protection Regulation. https://gdpr-info.eu/. Accessed 23 Oct 2024

2. Chee, F.Y.: Reuters. EU privacy watchdogs call for ban on facial recognition in public spaces (2021). https://www.reuters.com/technology/eu-privacy-watchdogs-call-ban-facial-recognition-public-spaces-2021-06-21/. Accessed 23 Oct 2024
3. Urgessa, T., Maeng, W., An, K.S., Lee, J.S.: Prototyping sensor-based people counting system. In: Proceedings of HCI Korea, pp. 121–126 (2016)
4. Li, H., Chan, E.C., Guo, X., Xiao, J., Wu, K., Ni, L.M.: Wi-counter: smartphone-based people counter using crowdsourced Wi-Fi signal data. IEEE Trans. Hum.-Mach. Syst. **45**(4), 442–452 (2015)
5. Navneet, D., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1. IEEE (2005)
6. Zhang, Y., et al.: Bytetrack: multi-object tracking by associating every detection box. In: European Conference on Computer Vision, pp. 1–21. Springer, Cham (2022)
7. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)
8. Li, Z., Liu, F., Yang, W., Peng, S., Zhou, J.: A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Trans. Neural Networks Learn. Syst. (2021)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
10. He, Q., Liu, K., Qu, L.: Efficient and robust indoor people detection based on RGB-D camera. In: 2017 Chinese Automation Congress (CAC), Jinan, China, pp. 1063–1068 (2017). https://doi.org/10.1109/CAC.2017.8242923
11. Yuan, Z., Zhang, Y., Duan, R.: RGB-D people detection and tracking from small-footprint ground robots. In: 2018 International Conference on Control and Robots (ICCR), Hong Kong, China, pp. 25–29 (2018). https://doi.org/10.1109/ICCR.2018.8534489
12. Tran, D.N.-N., Pham, L.H., Nguyen, H.-H., Jeon, J.W.: City-scale multi-camera vehicle tracking of vehicles based on YOLOv7. In: 2022 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia), pp. 1–4. IEEE (2022)
13. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, pp. 3645–3649 (2017). https://doi.org/10.1109/ICIP.2017.8296962
14. Tsai, W.-J., Huang, Z.-J., Chung, C.-E.: Joint detection, re-identification, and LSTM in multi-object tracking. In: 2020 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6. IEEE (2020)
15. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2961–69 (2017)
16. Yu, Y., Si, X., Hu, C., Zhang, J.: A review of recurrent neural networks: LSTM cells and network architectures. Neural Comput. **31**(7), 1235–1270 (2019)
17. Benahmed, H.K., Bensaad, M.L., Chaib, N.: Detection and tracking of honeybees using YOLO and StrongSORT. In: 2022 2nd International Conference on Electronic and Electrical Engineering and Intelligent System (ICE3IS), pp. 18–23. IEEE (2022)
18. Du, Y., et al.: Strongsort: make deepsort great again. IEEE Trans. Multimedia (2023)
19. You, X., Li, D., Zhang, M.: A top-view multiple people tracking system based on newest YOLOv5 and DeepSort using depth data. In: 2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), pp. 91–96. IEEE (2022)
20. Baek, J., Hong, S., Kim, J., Kim, E.: Efficient pedestrian detection at nighttime using a thermal camera. Sensors. **17**(8), 1850 (2017)
21. Altay, F., Velipasalar, S.: The use of thermal cameras for pedestrian detection. IEEE Sensors J. **22**(12), 11489–11498 (2022)

22. Kieu, M., Berlincioni, L., Galteri, L., Bertini, M., Bagdanov, A.D., del Bimbo, A.: Robust pedestrian detection in thermal imagery using synthesized images. In: 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy (2021)
23. Yun, S., Kim, S.: Recurrent YOLO and LSTM-based IR single pedestrian tracking. In: 2019 19th International Conference on Control, Automation and Systems (ICCAS), pp. 94–96. IEEE (2019)
24. Bellini, P., Palesi, L.A.I., Giovannoni, A., Nesi, P.: Managing complexity of data models and performance in broker-based internet/web of things architectures. Internet Things 100834 (2023)
25. Bellini, P., Bologna, D., Fanfani, M., Palesi, L.I., Nesi, P., Pantaleo, G.: Rapid prototyping & development life cycle for smart applications of internet of entities. In: 2023 27th International Conference on Engineering of Complex Computer Systems (ICECCS), pp. 142–151. IEEE (2023)
26. Adreani, L., Bellini, P., Fanfani, M., Nesi, P., Pantaleo, G.: Design and develop of a smart city digital twin with 3D representation and user interface for what-if analysis. In: Gervasi, O., et al. (eds.) Computational Science and Its Applications – ICCSA 2023 Workshops. ICCSA 2023. Lecture Notes in Computer Science, vol 14111. Springer, Cham (2023)
27. Alberti, F., et al.: Mobile mapping to support an integrated transport-territory modelling approach. Int. Arch. Photogramm. Remote. Sens. Spat. Inf. Sci. **48**, 1–7 (2023)
28. Kaehler, A., Bradski, G.: Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library. O'Reilly Media, Inc. (2016)
29. Ultralytics. Yolov5 github repository (2020). https://github.com/ultralytics/yolov5/tree/v6.0. Accessed 23 Oct 2024
30. Inui, A., et al.: Detection of elbow OCD in the ultrasound image by artificial intelligence using YOLOv8. Appl. Sci. **13**(13), 7623 (2023)
31. Collini, E., Palesi, L.A.I., Nesi, P., et al.: Flexible thermal camera solution for Smart city people detection and counting. Multimed Tools Appl **83**, 20457–20485 (2024). https://doi.org/10.1007/s11042-023-16374-x
32. Fanfani, M., Marulli, M., Nesi, P.: Addressing domain shift in pedestrian detection from thermal cameras without fine-tuning or transfer learning. In: 2023 IEEE International Conference on Smart Computing (SMARTCOMP), pp. 314–19. IEEE (2023)
33. Pei, Y., et al.: An elementary introduction to Kalman filtering. Commun. ACM **62**(11), 122–133 (2019)
34. Maggiolino, G., Ahmad, A., Cao, J., Kitani, K.: Deep OC-sort: multi-pedestrian tracking by adaptive re-identification. arXiv preprint arXiv:2302.11813 (2023)
35. Zhou, K., Yang, Y., Cavallaro, A., Xiang, T.: Omni-scale feature learning for person re-identification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3702–3712 (2019)
36. Bashir, F.I., Khokhar, A.A., Schonfeld, D.: Object trajectory-based activity classification and recognition using hidden Markov models. IEEE Trans. Image Process. **16**(7), 1912–1919 (2007)
37. Chen, T., Guestrin, C.: Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–94 (2016)
38. Bian, J., Tian, D., Tang, Y., Tao, D.: Trajectory data classification: a review. ACM Trans. Intell. Syst. Technol. (TIST) **10**(4), 1–34 (2019)
39. Jakkula, V.: Tutorial on support vector machine (SVM). School of EECS, Washington State University **37**(2.5), 3 (2006)
40. Steinley, D.: K-means clustering: a half-century synthesis. Br. J. Math. Stat. Psychol. **59**(1), 1–34 (2006)

41. Jia, X., Zhu, C., Li, M., Tang, W., Zhou, W.: LLVIP: a visible-infrared paired dataset for low-light vision. arXiv (2021)
42. Luiten, J., et al.: Hota: a higher order metric for evaluating multi-object tracking. Int. J. Comput. Vision **129**, 548–578 (2021)
43. Kontopoulos, A.M., Zissis, D., Tserpes, K.: A computer vision approach for trajectory classification. In: 2021 22nd IEEE International Conference on Mobile Data Management (MDM), Toronto, ON, Canada, pp. 163–168 (2021). https://doi.org/10.1109/MDM52706.2021.00034
44. Lin, S., et al.: ColorByte: a real time MOT method using fast appearance feature based on ByteTrack. Null, pp.1–6 (2022). https://doi.org/10.1109/CBD58033.2022.00010
45. Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M.: Optuna: a next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2019)
46. Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. In: Proceedings Advance Neural Information Processing System, pp. 4768–4777 (2017)