Understanding People Flow in Transportation Hubs

João Carvalho⬝, Manuel Marques⬝, and João Paulo Costeira⬝

Abstract—In this paper, we aim to monitor the flow of people in large public infrastructures. We propose an unsupervised methodology to cluster people flow patterns into the most typical and meaningful configurations. By processing 3-D images from a network of depth cameras, we build a descriptor for the flow pattern. We define a data-irregularity measure that assesses how well each descriptor fits a data model. This allows us to rank flow patterns from highly distinctive (outliers) to very common ones. By discarding outliers, we obtain more reliable key configurations (classes). Synthetic experiments show that the proposed method is superior to standard clustering methods. We applied it in an operational scenario during 14 days in the X-ray screening area of an international airport. Results show that our methodology is able to successfully summarize the representative patterns for such a long observation period, providing relevant information for airport management. Beyond regular flows, our method identifies a set of rare events corresponding to uncommon activities (cleaning, special security, and circulating staff).

Index Terms—People flow monitoring, unsupervised clustering, depth cameras.

I. INTRODUCTION

IN THIS article, we address the problem of crowd monitoring in public infrastructures. We propose a sensing and data processing framework which captures crowd occupancy in these large spaces, and identifies and classifies it into meaningful spatial patterns/classes. We tested it in a real-life scenario in the main security X-ray screening area at Lisbon's international airport (LIS), Portugal.

Airports are transportation hubs subject to strict service-level agreements (SLA), high security risks and high operational costs. These factors put great pressure on human and physical resources, calling for tight monitoring of passenger flow within the infrastructure. In particular, the security and identification checkpoints are critical bottlenecks in the path between the check-in and the departure gate. Besides the risk of SLA violation, these bottlenecks have great impact on operational and commercial costs. Our work focuses on characterizing how passengers flow while waiting for inspection in these critical checkpoints.
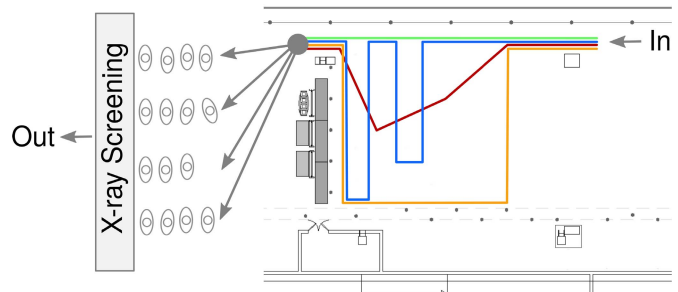
Fig. 1. X-ray screening queue area at the Lisbon international airport. The colored lines illustrate examples of paths in the queue area. The queue area is around $200m^2$.

The sensing infrastructure we propose consists of a network of depth cameras that provide 3D data of the covered space. A set of pre-processing algorithms anonymously detect people in the 3D point cloud and compute the 3D space occupancy map. Fig. 1 displays the specific geometry of LIS X-ray area,[1] where passengers enter from the right, wait for their turn and exit to the left towards individual X-ray booths. In such a large space—more than $200m^2$—several configurations emerge, shaped by inflow variations, processing capacity, planning protocols and particular events.

Currently, the global status of the X-ray system is monitored using global counts of in-out passenger flow [1], [2]. These global measures do not account for the internal state of the queue that reflects the instantaneous operating condition of the whole system. Queue modeling (e.g., flow parameterization) is very difficult to do, despite some structure imposed by queue guides and a relatively controlled environment. X-ray queue configurations are set in an ad-hoc manner and often change according to discretionary decisions by operations personnel. Also, innumerable "small" local decisions impact the global system: metal detectors settings, protocols for suspicious luggage screening, passenger inspection or security personnel skills. All these aspects are hard to account for but affect the performance and overall state of the X-ray queue.

Our methodology for classification of passenger flow patterns identifies meaningful classes that encode space occupancy. Classes are defined by space occupancy patterns that appear regularly over time (*regular* patterns) as well as sporadic patterns that correspond to "atypical behaviours" (*irregular* patterns). The classes and its number are not known *a priori* and the amount of outliers can be significative. Because of the large amount of data, it is very difficult for the airport management to define the most relevant classes.

[1]The physical infrastructure described here no longer exists.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

2　　　　　　　　　　　　　　　　　　　　　　　　　　　　IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS

Building on previous work in the computer vision domain [3], we encode regularity/irregularity through linear subspace models that explicitly represent these regularity concepts [4], [5]. Such subspaces are determined from data in an unsupervised way.

Our methodology grants decision-makers with the much sought holistic measure of the "operational state" of the infrastructure. The methodology is general and applicable to a wide variety of public infrastructures such as railway stations or commercial shopping centers.

In summary, our main contributions are:

- a 3D sensing infrastructure for queue data collection;
- a descriptor computed from 3D data of multiple cameras that captures queue patterns;
- an algorithm that explicitly models and identifies *regular* and *irregular* flow patterns and clusters the regular data into classes of "typical" operation modes.

By identifying regular "operating states" of the whole system, airport experts are able to rank and map them to desirable performance indicators. Having these states, one can build individual and more accurate models for each of them using queueing theory, for example, seeking to improve management and security. The ultimate goal is to increase costumer satisfaction. This interaction with the management and operational staff is very important in order to understand the behavior of the whole system [6]. Our methodology can be understood as a pre-processing stage whose information can feed the planning, scheduling and performance analysis systems of the airport.

II. PREVIOUS WORK

In complex environments such as an airport [7], the flow of passengers depends on a large number of variables: staff performance, passenger behaviour, hand luggage contents, sensitivity of metal detection systems, among others. Traditional approaches based on discrete event systems [8], [9] or queueing theory [10]–[12] model airport activity as a function of several such variables. However, these methods are difficult to tune because they are highly dependent on variables that are unobservable or modeled with (unrealistic) stationary distributions (*e.g.*, arrival and service rates). Also, these are based on data manually acquired within limited time frames or based on aggregated statistics. Besides providing tools for real time data acquisition, our methodology identifies the stationary states of the queue, allowing the design of individual and more accurate models for each state.

To automatically acquire people flow data, several works track and count people with a single video camera [13], [14] or a single over-the-head RGB-D camera [15]–[17].

In crowded scenes, instead of tracking people individually, several works propose solutions for activity detection or estimation of the number of people present in the scene. For example, Convolutional Neural Networks (CNN) are employed in [18] to compute a per-pixel crowd counting map in order to estimate the number of people crossing a line. Reference [19] improves crowd counting by training a CNN using two related objectives, crowd density and count, to obtain a better local optimum for both. Alternatively [20], proposes a multi-column

CNN to map an image to its crowd density map. Ma and Chan [21] and Mukherjee *et al.* [22] use features derived from optical flow to train a system for counting people in regions of interest at crowded places. The optical acceleration and histogram of optical flow gradients can be combined to detect abnormal objects or speed violation in pedestrian scenes [23]. A probabilistic approach is employed by Wang *et al.* [24], to cluster moving pixels and video segments in order to model atomic activities and interactions in crowded places. In [25], the detection of abnormal behaviors is done with a spatio-temporal analysis of a densely sampled video volumes.

Clustering individual trajectories is another common approach to analyze the flow of people. In [26], feature point tracks are clustered and dominant trajectories are identified by fitting polynomials to cluster mean points. Approach [27] proposes the Robust K-means algorithm for clustering data that is less sensitive to the initialization of the K clusters. K-means and agglomerative clustering are used in [28] to cluster trajectories into common patterns. Alternatively, in [29], trajectories are clustered with Fuzzy C-means and modeled with HMM, for trajectory analysis. Also, local distance and similarity measures are frequently employed to cluster and analyze trajectories and flow data [30], [31].

Wu and Radke [32] propose a system for monitoring queues by tracking people using a network of over-the-head video cameras. A solution to integrate existing monitoring technology is proposed in [1]. It studies methodologies for people counting and individual tracking for queue monitoring and behavior analysis.

In our scenario, the placement of the sensors is limited by the infrastructure, leading to large scene perspective distortion, people occlusion and rendering the registration of video cameras unfeasible. Moreover, due to privacy concerns, the use of video cameras is not allowed in many locations, including the security checkpoint at Lisbon Airport. This precludes the use of the referenced approaches, which, besides using RGB cameras, assume: high ceilings, proposing over-the-head solutions [15], [16], [32]; access to airport sensing infrastructure [1]; existence of reliable individual trajectories [26], [28], [29], [33]; absence of outliers , as irregular patterns [27]; and large amounts of training data [18]–[20].

To the best of our knowledge, there is no work proposed for unsupervised identification and classification of people flow patterns from depth data. Our approach is applicable in a large range of scenarios because it does not require a tracking scheme. Also, we bring the novelty of testing our methodology with 14 days of real data acquired at an international airport.

III. SENSING INFRASTRUCTURE AND DATA DESCRIPTOR

To characterize the X-ray queue area, we want to capture the occupancy pattern of the passengers over time. As we will see in Section IV, counting people at the entrance or exit of the queue is not enough to understand the state and shape of the queue. Also, individually tracking people in such a large area is unreliable, due to frequent occlusion of passengers. We propose a descriptor that captures the dynamics of the space and is robust to misdetections. We present here a sensing

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

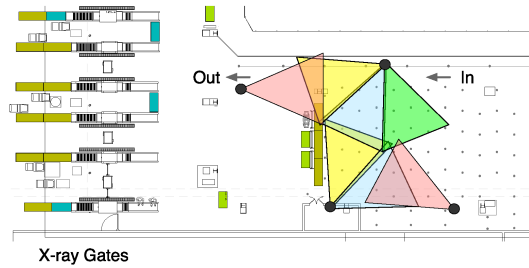CARVALHO *et al.*: UNDERSTANDING PEOPLE FLOW IN TRANSPORTATION HUBS

3

Fig. 2. Camera setup in the X-ray queue with corresponding field of view (FOV). Each colored triangle represents the FOV of a different camera. Passengers enter from the right and exit on the left.
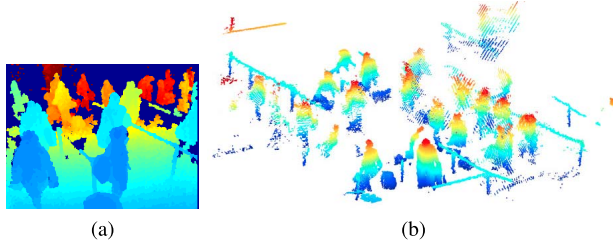

Fig. 3. Depth image and 3D representation: (a) Depth image from one of the cameras placed at the X-ray queue area at the Lisbon International Airport. The value of each pixel represents the depth of the object rather than its brightness or color. The colors are indexed to the distance to the camera, from blue (closer) to dark red (further). Dark blue represents areas not visible by the camera. (b) 3D representation of the space. Colors range from dark blue, for points closer to the ground plane, to dark red, for points further from the ground plane.

infrastructure and, based on 3D data processing, the proposed descriptor.

Fig. 2 depicts the network of depth cameras installed to cover the final stage of the queue area. Cameras are calibrated and 3D point clouds are registered into a global reference frame. So, from depth images provided by the seven cameras (Fig. 3a), we build a global 3D representation of the space (Fig. 3b). Since we have a noisy sensor and multiple objects in the foreground, we used a procedure we developed in [34] to detect people in 3D point clouds (Section III-A). Using the results of this detection procedure, we compute a ground occupancy map and use it as a descriptor for the passenger flow (Section III-B).

A. People Detection

The method takes a 3D point cloud as input (Fig. 3b) and outputs the 3D points and centroids corresponding to each person. Fig. 4a shows the 3D point clouds classified as "person", with one color per person, and Fig. 4b shows the centroid of each person projected on the ground plane.

The 3D space is scanned with a fixed size box and the point cloud falling within that volume is classified as *person* or *not person*. The classifier, based on random trees, was trained with point clouds including people in different poses relative to the camera. From each candidate point cloud, we compute its height, volume and the area occupied by the projection into the ground plane. The output of this method is the labeling of the point cloud and the corresponding centroids in ground
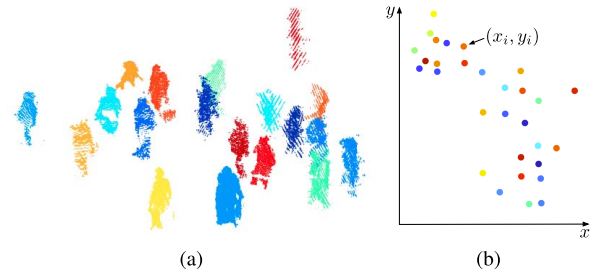

Fig. 4. People detection method results: (a) point cloud of the people detection, with one color per person; (b) centroids (x, y) of each detection.
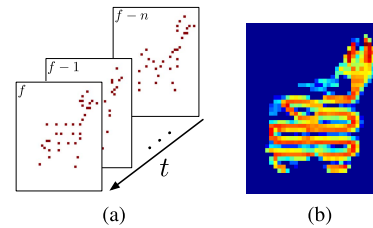

Fig. 5. Occupancy maps: (a) buffer of binary maps with dark red cells if occupied and white otherwise; (b) occupancy map for $5min$ sized buffer. The colors range from dark blue, for the null value, to dark red, for the maximum value of the map.
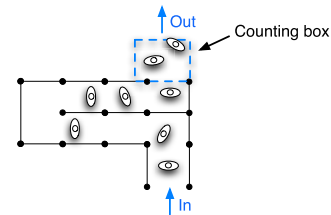

Fig. 6. Illustration of a counting box at the exit of a queue.

coordinates, (x_i, y_i) for person i, as shown in Fig. 4. For further details, including training process and performance, see [34].

B. Queue Pattern Descriptor

We characterize the shape of the pedestrian flow using an occupancy map as data descriptor for the queue state. We build this occupancy map from the centroids (x_i, y_i) output from the people detection method. The floor is discretized into a fixed size grid and we obtain a binary map where each cell of the map is set to 1 if occupied and 0 otherwise. A cell is occupied if a centroid falls into it. Fig. 5a shows a set of these binary maps, from frame $f - n$ to f, where each dot corresponds to a occupied cell in the map.

The occupancy map of a given time period is the spatio-temporal average of the set of binary maps for that same period.[2] We could apply other filters to the buffer of binary maps, however, we found that using the spatio-temporal average successfully captures the shape of pedestrian flow. Fig. 5b shows an example of an occupancy map. This map is a vector in $\mathbb{R}^d$ that evolves in time, depending on the space usage, where d is the number of spatial cells of the map.

[2]A map can be instantaneous or integrated into longer periods, such as $30s$, $1min$, $5min$, $1h$.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

4

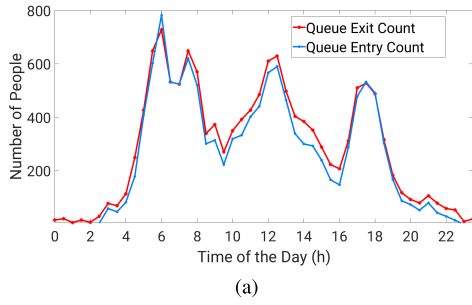IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS

Fig. 7. Number of people passing by the X-ray queue detected with our approach versus ticket validation count. Differences between the two are due to the ground truth being based in boarding pass validation at the entrance of the queue and the count being relative to all the people, including staff, at the exit of the queue. Our detection methodology is able to capture the outflow variation.
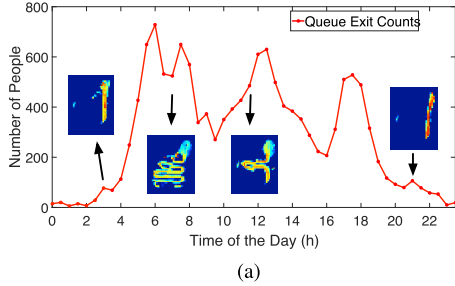

Fig. 8. X-ray exit count versus occupancy maps. Similar counts may correspond to different queue configurations.

IV. QUEUE PATTERN VERSUS THROUGHPUT

In this section, we show that using passenger counts [1], [2], is not enough to characterize the occupancy of the area being monitored.

Based on the people detection procedure, we are able to count people passing in a small area at the exit of the queue (Fig. 6). In this scenario, we have 4% error, similar to the classification error obtained in the original paper [34]. Fig. 7 shows the number of passengers at the entrance (blue line) and at the exit (red line) of the queue during one day. The passenger count at the entrance is provided by the boarding pass scanning system. However, at the queue exit, our system counts people without distinction between passengers and staff, therefore, the difference between the two counts is due not only to the detection error, but also due to staff passing the exit and the delay between the two counting systems (one is at the entrance and the other at the exit). Despite these sources of error, we can see in Fig. 7 that our detection procedure successfully captures the outflow of the queue. However, the count of passengers is not sufficient to identify the queue state.

Fig. 8 shows four occupancy maps corresponding to four time instants. The periods around $07:00\,am$ and $12:00\,pm$ have very similar counts, however, the queue configurations are different. The state of the queue is dependent on many factors, including, for example, the relation between the queue inflow, service performance and the number of X-ray operating gates. Since these indicators are not available, the outflow of the queue is not enough to determine the queue/service state. Our solution relies on the occupancy map to capture the usage of the space.
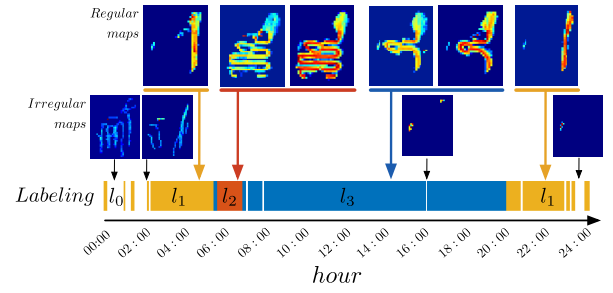

Fig. 9. This figure illustrates the goal of out methodology: going from occupancy maps to labels corresponding to the different queue configurations. Here, we illustrate three regular clusters (l_1 to l_3) and a cluster with irregular patterns (l_0, white label). Each cluster is composed by several similar patterns.

V. DETECTING QUEUE CONFIGURATIONS

In the previous sections, we described our sensing setup composed by a network of depth cameras that provides 3D data. With this data, we are able to detect people and compute an occupancy grid map – a descriptor that does not rely on tracking to encode people movements.

With the occupancy maps for a given period of time, we aim to cluster them into representative classes, where each class corresponds to a different queue configuration. Fig. 9 illustrates this goal, with similar patterns grouped in the same cluster. Each cluster is represented in different color and with label l_i, where i is the class index. In this example, we illustrate three clusters with regular patterns (l_1 to l_3) and a white label (l_0) for irregular patterns.

Since we do not know the classes *a priori*, a supervised approach is not adequate. On the other hand, the traditional unsupervised clustering methods do not guarantee a solution to our problem, as they require the number of classes to be known *a priori*, and do not deal with outliers. As Fig. 9 shows, we may have a considerable percentage of outliers in the data (white label, l_0).

We model a typical pattern as a linear combination of other maps in the same class and compute the combination coefficients by solving a convex optimization problem. Because typical patterns appear regularly over time, we draw inspiration from the *self-expressiveness* property [4], [5], which states that a point in a subspace can be represented as a linear combination of points in the same subspace. So, each class is modeled as a subspace, where each pattern of a given class is a point in the corresponding subspace. Solving this convex optimization problem, we have a measure of irregularity that assesses how distant a pattern is from the linear model. If this measurement is too high, the pattern is considered irregular (outlier).

Our method consists of two main steps: *identifying regular and irregular maps*; and *clustering regular maps*. Fig. 10 summarizes the methodology.

A. Identifying Regular and Irregular Maps

Consider the data matrix

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_m \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{d \times m}, \quad (1)$$

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

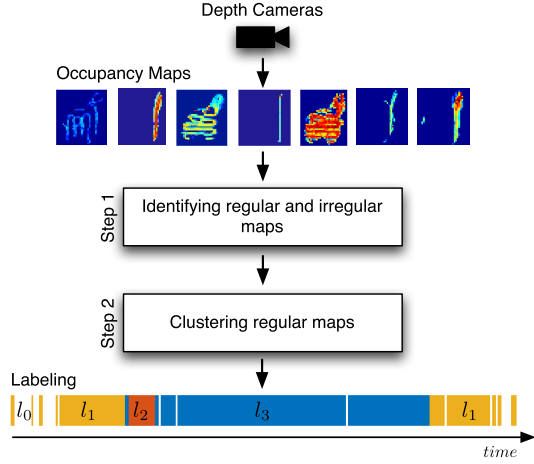CARVALHO *et al.*: UNDERSTANDING PEOPLE FLOW IN TRANSPORTATION HUBS

5

Fig. 10. Diagram summarizing the two main steps of the proposed queue classification methodology.

where column $\mathbf{x}_i$ is a vectorized occupancy map, d is the data dimension and m is the number of maps.[3]

A map $\mathbf{x}_i$ is represented by a convex combination of other maps, being modeled as

$$\mathbf{x}_i = \sum_{j=1, j \neq i}^{m} \mathbf{x}_j c_{ji} \qquad (2)$$

$$\sum_{j=1, j \neq i}^{m} c_{ji} = 1, \quad c_{ji} \geq 0$$

where c_{ji} is the coefficient that relates map j with map i. The convex combination ensures that the map $\mathbf{x}_i$ is in the convex hull of other maps and, because of $c_{ji} \geq 0$, that the maps in the convex hull have similarities with $\mathbf{x}_i$. Because a map should be expressed as a combination of maps belonging to the same cluster/subpace, our goal is to find the subspaces that best explain all the data. In matrix form, the model is given by

$$\mathbf{X} = \mathbf{X}\mathbf{C} \qquad (3)$$

with

$$\mathbf{1}_m^T \mathbf{C} = \mathbf{1}_m^T, \quad \mathbf{C} \geq \mathbf{0}_{m \times m}, \quad diag(\mathbf{C}) = 0$$

where $\mathbf{C} \in \mathbb{R}^{m \times m}$ is the coefficients matrix and $\mathbf{1}_m$ is a vector of ones with dimension m. Imposing the null diagonal of $\mathbf{C}$, $diag(\mathbf{C}) = 0$, excludes the trivial solution of a map explaining itself. The irregularity of a map is measured by how much the reconstruction of that map violates the linear model. This is given by

$$\mathcal{I}_i = \|\mathbf{x}_i - \mathbf{X}\mathbf{c}_i\|_1, \qquad (4)$$

where the operator $\|.\|_1$ is the ℓ_1-*norm* and $\mathbf{c}_i$ is the i-th column of $\mathbf{C}$. Our goal is to find the convex combination that best reconstructs each map, in the sense of minimizing the irregularity measure. Then, we wish to generate map $\mathbf{x}_i$

[3]Bold capital letters, $\mathbf{A}$, represent matrices. Bold lower-case letters, $\mathbf{a}$, represent column vectors. Bold lower-case letters with subscript, $\mathbf{a}_i$, represent the ith column of matrix $\mathbf{A}$. Scalars are denoted by non-bold lower-case letters, a. The scalar element in row j and column i of matrix $\mathbf{A}$ is denoted by a non-bold lower-case letter with two subscripts, a_{ji}.
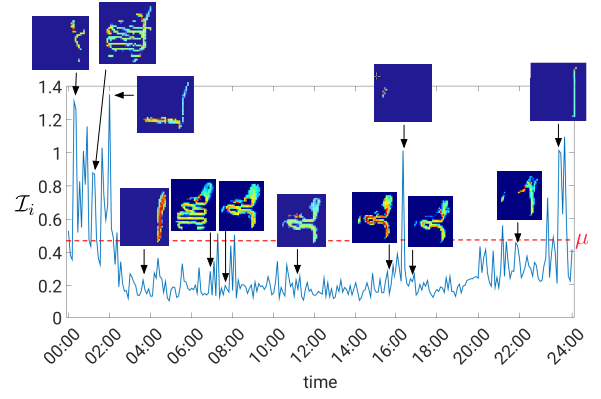

Fig. 11. Irregularity measure for a complete day. Some of the corresponding occupancy maps are presented to show that higher $\mathcal{I}_i$ values correspond to more peculiar patterns.

with a small number of maps. By using the ℓ_1-*norm* in (4), we are accounting for sparse error in the data.

The coefficients are obtained by solving the following optimization problem

$$\min_{\mathbf{C}} \mathcal{I} \qquad (5)$$
$$\text{s.t. } diag(\mathbf{C}) = 0$$
$$\mathbf{1}_m^T \mathbf{C} = \mathbf{1}_m^T$$
$$\mathbf{C} \geq \mathbf{0}_{m \times m},$$

where

$$\mathcal{I} = \|\mathbf{X} - \mathbf{X}\mathbf{C}\|_1 \qquad (6)$$

is the global irregularity measure of the data and operator $\|.\|_1$ is the L_1 matrix norm. The solution is computed in parallel by solving for each $\mathbf{c}_i$ separately [3].

By solving (5) we generate the map $\mathbf{x}_i$ with the other maps in $\mathbf{X}$. However, when the error $\mathcal{I}_i$ is too large, a map is an outlier (a unique pattern). Therefore, the irregularity $\mathcal{I}_i$ is in fact a measure of uniqueness of the map $\mathbf{x}_i$ within the data set. Fig. 11 shows the irregularity measure, $\mathcal{I}_i$, for m occupancy maps corresponding to a full day. As expected, maps with lower $\mathcal{I}_i$ are maps capturing typical states of the queue. On the other hand, large values correspond to the occasional patterns.

Choosing a value of p, we label as *irregular* the $p\%$[4] of maps with highest irregularity and the others as *regular*.[5] With the split and merge approach, presented in next section, the methodology is robust to a large range of p values.

B. Clustering Regular Maps

Solving (5), we are able to identify regular and irregular (outlier) patterns. Filtering out outliers, the data can be clustered using common clustering methods. Here, we take advantage of the sparse coding framework. Note that $\mathbf{C} + \mathbf{C}^T$ defines an undirected graph where the nodes represent the maps and edges represent association (non-null coefficients) between maps. Because, the number of classes K is unknown *a priori*, we estimate K by estimating the number of zero

[4]This is equivalent to apply a threshold μ to $\mathcal{I}_i$ [3].
[5]Note that in [3] these irregular observations correspond to physical keypoints of a shape/model. Whereas here we interpret them as maps corresponding to unique patterns in the data.
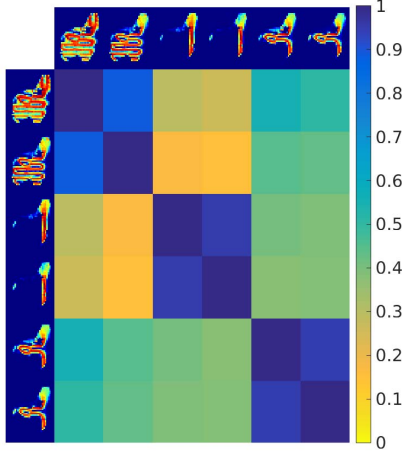
Fig. 12. NSI affinity matrix for six exemplifying clusters. For each cluster, we show a map belonging to it. Pairs of clusters corresponding to similar configuration have NSI value close to 1.

eigenvalues of the Laplacian of this graph. However, small clusters (with few points) have small impact in the eigenvalues and, when such clusters exist, K is underestimated. To retrieve the smaller clusters, we segment the data into a larger number, γ, of linear subspaces and use a measure intrinsic to the model to compute the distance between the subspaces. If this distance is small, clusters are merged.

The criterion we use is the Normalized Subspace Inclusion (NSI), a distance between subspaces of any dimension [35]. Formally, it is defined as

$$NSI(\mathcal{L}_1, \mathcal{L}_2) = \frac{tr(\mathbf{U_1^T U_2 U_2^T U_1})}{\min(d_1, d_2)}, \tag{7}$$

where $\mathcal{L}_1$ and $\mathcal{L}_2$ are linear subspaces in $\mathbb{R}^n$, $\mathbf{U}_1$ and $\mathbf{U}_2$ are their orthonormal basis and d_1, d_2 are the subspaces dimensions. This criterion measures inclusion of subspaces, generalizing the angle between two subspaces. Then, choosing this criterion comes naturally from the assumptions we made previously, that the data lies in the union of linear subspaces. Similarly, and to exploit this subspace structure in all the clustering process, we over-segment the regular data using *Spectral Clustering* with $\mathbf{C} + \mathbf{C^T}$ as adjacency matrix.

Fig. 12 shows an affinity matrix with the NSI for six clusters. For each cluster we show here only one of its maps. It is clear that the NSI value between clusters that correspond to very similar configurations is close to 1.

Clusters are merged if their pair-wise NSI value is above a threshold. Similarly, after this step, we reassess the maps labeled as *irregular*. We compute the pair-wise NSI affinity for each irregular map and cluster. The map is unified to the cluster with larger affinity if the value is above a threshold, otherwise, the maps remain labeled *irregular*. This consolidation step, allows a large range for $p\%$ in the selection of irregular/regular patterns.

C. Methodology Summary

The methodology proposed in this section is summarized in Algorithm 1. First, we solve (5) to compute the coefficients that relate the patterns (line 1) and then we rank the maps

by irregularity to identify regular and irregular maps (line 2). Next, we estimate the number of clusters and partition regular data accordingly (lines 3 and 4). Then, clusters are consolidated using the affinity measure (lines 5 and 6). Finally, irregular maps are classified according to the consolidated clusters (line 7). The output is the labeling for all data points and the corresponding partition into regular and irregular patterns.

Algorithm 1 Identifying and Clustering Regular Occupancy Maps in the Data

---

Input: X - Data matrix
   p - % of maps to label as irregular
   th - NSI threshold
Output: $\mathbf{X}_{reg}$ - Regular data
   $\mathbf{X}_{irreg}$ - Irregular data
   cl_{reg} - Regular data class labels

Step 1: Identifying Regular and Irregular Maps
1: $\mathcal{I} \leftarrow$ Solve (5) for **C**
2: $\mathbf{X}_{reg}, \mathbf{X}_{irreg} \leftarrow$ Divide regular and irregular data
                     according to p and $\mathcal{I}$

Step 2: Clustering Regular Maps
3: $\gamma \leftarrow$ Estimate number of clusters
4: $cl_{reg} \leftarrow$ Segment $\mathbf{X}_{reg}$ into γ clusters
5: $affinity \leftarrow$ Compute NSI between γ clusters
6: $cl_{NSI} \leftarrow$ Merge clusters in cl_{reg} with affinity $> th_{NSI}$
7: $cl_{reg}, \mathbf{X}_{reg}, \mathbf{X}_{irreg} \leftarrow$ Reclassify $\mathbf{X}_{irreg}$ according
                     to cl_{NSI}

Next, we present results of applying this methodology to real data.

VI. EXPERIMENTAL RESULTS

In this section, we present quantitative results for synthetic data and qualitative results for real data, with periods of one day and fourteen days. We used seven depth cameras to cover the X-ray queue area at the Lisbon international airport. The descriptor used integrates $5min$ occupancy.

A. Synthetic Data

To quantitatively assess the performance of our method and to compare it with other clustering methods, we perform a set of experiments with synthetic data. The classes are unknown to the algorithm and the data is corrupted with noise and outliers.

We consider a squared space of $400m^2$, with a total of 8 doors and several possible paths between those doors (Fig. 13a). At each time instant, we randomly generate a simulated passenger at a given door. Each passenger moves along a set of predefined paths (subject to perturbations), with speed $s \backsim \mathcal{N}(1.4ms^{-1}, 0.05)$. Similar to the method for people detection used in the real setup [34], objects are correctly detected 96% of time. Outlier trajectories are created with passengers partially following one of the regular paths but then temporarily moving away to go through random points in
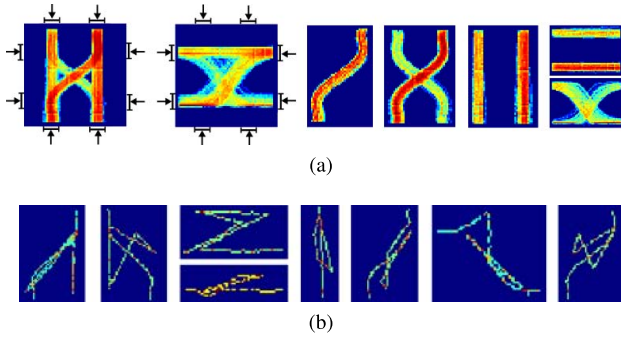
Fig. 13. Examples of occupancy maps generated from synthetic data: (a) regular maps, with the 8 doors illustrated in the two left maps; (b) outlier maps. Each map integrates data from a period of $5min$.

TABLE I

AVERAGE CLUSTERING ERROR AS A FUNCTION OF THE PERCENTAGE OF OUTLIERS, p_{out}. FOR EACH TRIAL WITH K-MEANS AND K-MEDOIDS, WE USED THE BEST OF 10 REPLICATES. FOR OUR METHOD, WE REPORT THE ERROR FOR $p = 0.50$, NSI THRESHOLD 0.93 AND $\gamma = 17$ CLUSTERS

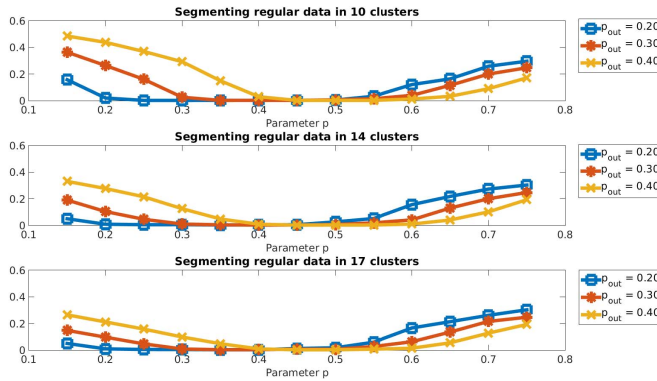| p_{out}	0.20	0.30	0.40	0.50	0.60
K-Means	0.284	0.334	0.388	0.393	0.402
DBSCAN	0.206	0.223	0.228	0.223	0.212
K-Medoids	**0**	0.030	0.030	0.067	0.104
Proposed Method	0.022	**0.009**	**0.004**	**0.003**	**0.005**

Fig. 14. Clustering error versus parameter p for three levels of outliers. Top plot: segmenting data in 10 clusters. Middle plot: segmenting in 14 clusters. Bottom plot: segmenting in 17 clusters. In each plot we show the error for three percentages of outliers, $p_{out} \in \{0.20, 0.30, 0.40\}$.

space, returning finally to the normal course. We build $5min$ occupancy maps as explained in Section III-B, with one binary map per second. Fig. 13 shows some examples of regular and irregular occupancy maps created with this synthetic data. These data recreates the real setting, with a similar large space with several doors and paths, passengers moving at the average human speed and occupancy maps with the same characteristics. In the following experiments, we create data matrices with 10 clusters, corrupted with several percentages of outliers. For each such percentage, we run 20 simulated experiments with 576 maps each.

We compare our method with three well established clustering approaches: *K-Means* [36], *K-Medoids* [37] and DBSCAN [38]. We include K-Means because it is widely used, however, it is inadequate for this application, performing poorly. We evaluate the clustering error, defined as the rate
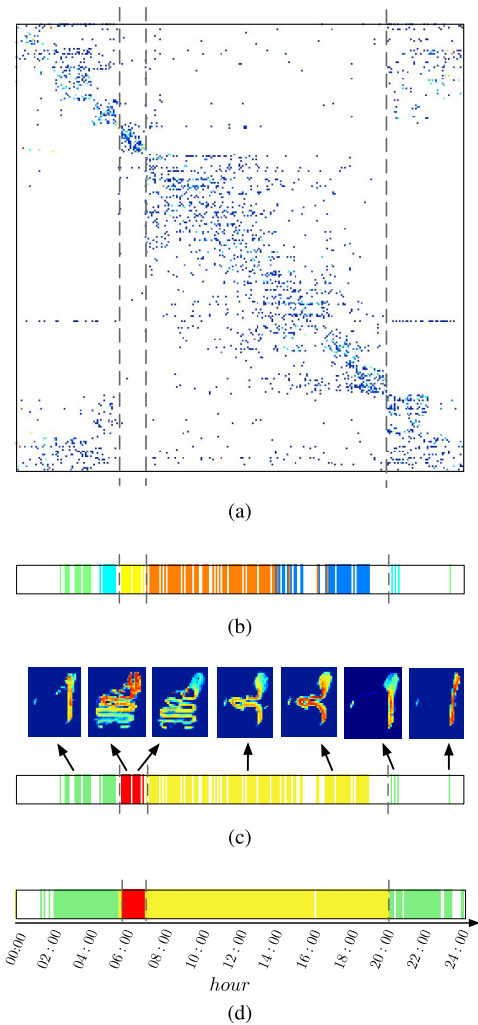

Fig. 15. Results for one day of data: (a) coefficients matrix $\mathbf{C}$; (b) over-segmentation after labeling 50% of the maps as irregular; result for $\gamma = 5$, with one color per cluster (white for irregular maps); (c) consolidated clusters obtained using the NSI criterion, with one color per cluster, and some of the corresponding regular maps; (d) final clusters, obtained after classifying maps initially labeled as irregular. The gray vertical dashed line in all sub-figures allows to relate coefficients in $\mathbf{C}$ with clustering results.

between incorrectly classified points and the total number of points. In this comparison, we consider only the error in the regular maps, meaning we do not assess the explicit identification of irregular maps. We input the correct number of clusters to K-Means and K-Medoids, and use angle between points as distance function. Table I shows the average clustering error as a function of the percentage of outliers. DBSCAN accounts for outliers in the data but, similar to K-Means, performs poorly with this data. K-Medoids achieves small errors for lower percentages but loses accuracy as the number of outliers increases. On the other hand, our approach is able to achieve error close to zero for all percentages of outliers. Note that our error is higher with 20% of outliers because we set $p = 0.50$ and, for some experiments, that value of p removes classes from the regular data. With a smaller p, we can also achieve zero error for this amount of outliers.

In Fig. 14, we evaluate the clustering error as a function of the parameter p, percentage of outliers and as function of number of clusters in which the regular maps are seg-

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

8   IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS
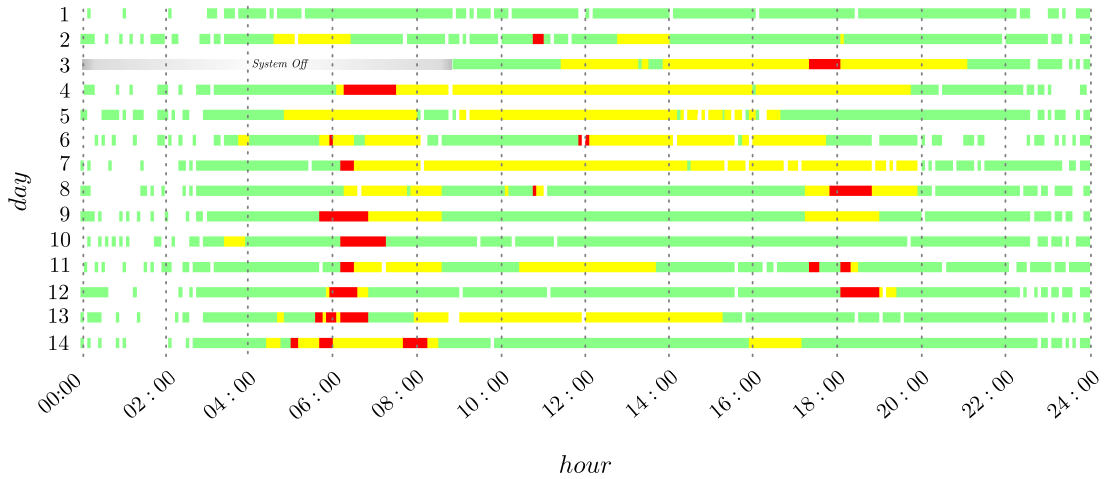

Fig. 16.   Results of our methodology for 14 days of data, with $\mathbf{X} \in \mathbb{R}^{d \times 3583}$. Days are shown separately for easier analysis. Each queue configuration is labeled with a different color, with white corresponding to irregular periods. Day 3 has a large period without labeling, identified in gray, because the acquisition system was off.

mented, γ. As we show, our method has the best performance for a large range of values of parameter p. To achieve this, we must ensure that all outliers are removed, $p > \%outliers$, and that none of the clusters is removed from the regular data. On the other hand, even if not all outliers are removed, the split and merge strategy improves the results because the remaining outliers have a reduced impact on the original clusters.

B. Real Data: Analyzing One Day

Here we present in detail the results of each step of our methodology applied to real data of one day. Fig. 15a shows the $\mathbf{C}$ matrix obtained by solving (5) for this day. The block diagonal shape of $\mathbf{C}$ suggests that classes appear in sequence and do not change often during the day.

Fig. 11 shows the *irregularity* for this sequence of maps, given this matrix $\mathbf{C}$. After ranking the maps, we remove the 50% with highest irregularity. Next, we partition the *regular* data (50% with lowest error) in 5 clusters using spectral clustering.[6] Fig. 15b shows the obtained labels, with one color per cluster and irregular maps in white. By consolidating these clusters with the NSI criterion, we obtained three clusters (green, yellow and red labels) corresponding to three different queue configurations, depicted in Fig. 15c. Finally, we classified the 50% maps previously labeled as irregular to assess if they belong to any regular clusters. Fig. 15d shows the final labels, where 11% of the maps remain labeled *irregular*.

During the periods 00:00*am* - 06:15*am* and 07:45*pm* - 11:59*pm*, the queue had low inflow of passengers and people would go straight from the entrance to the exit of the queue (green label in Fig. 15d). The period with largest passenger inflow occurs between 06:15*am* and 07:15*am* (red label). In this period, the queue structure is more complex, occupying a larger area to accommodate all passengers. Yellow label corresponds to a configuration with moderate flow. Irregular

maps, with white label, occur mainly in the night periods (with low passenger inflow), when cleaning and maintenance operations are performed.

C. Real Data: Analyzing an Extended Time Period

Fig. 16 shows results for 14 days of operation. The color labeling is shown for each day separately, although we applied the methodology to all data, with $\mathbf{X} \in \mathbb{R}^{d \times 3583}$. These colors correspond to the same configurations of Fig. 15. White label is associated with irregular or empty maps.[7]

Although all days are different from each other, periods with more inflow look similar: between 06:00*am* and 07:00*am* or between 06:00*pm* and 07:00*pm*. Irregular periods of time, as the ones depicted in Fig. 17, occur mainly in the early or late hours of the day and represent 12% of the fourteen days, a time lapse of around 39 hours.[8] Without identifying the outliers, the clusters computed by common approaches are contaminated and the corresponding subspaces may not represent the meaningful classes.

As we previously showed in Fig. 8, similar passenger count at specific periods corresponds to different queue patterns. Fig. 18 shows the same occurs for full days. This is due to several factors affecting the state of the queue, such as the number of open X-ray gates, number of staff members operating or luggage contents, for example. In other words, for the same number of passengers, the service can be operated differently.

VII. Discussion

We propose an unsupervised approach with few parameters to setup. The first parameter is the period of time covered by each occupancy map, *e.g.*, 30*s*, 1*min*, 5*min*. This period of time should be chosen depending on the activity we want to

---

[6]The adjacency matrix for the Spectral Clustering is $\mathbf{C} + \mathbf{C}^T$.

[7]Empty maps were not included in the data matrix.

[8]Day 3 has a large period of time without labeling because the acquisition system was off.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

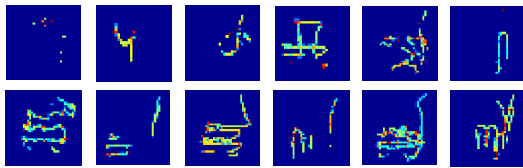CARVALHO *et al.*: UNDERSTANDING PEOPLE FLOW IN TRANSPORTATION HUBS

9

Fig. 17.   Some of the irregular maps identified within a 14 day period.
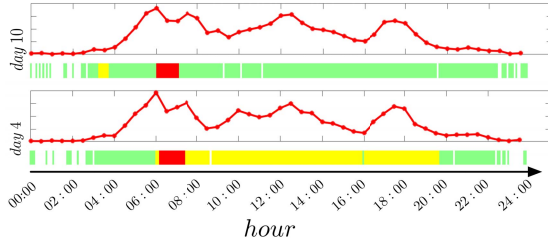

Fig. 18.   Counting and labeling for two different days. Although the counting is similar, the days are different.
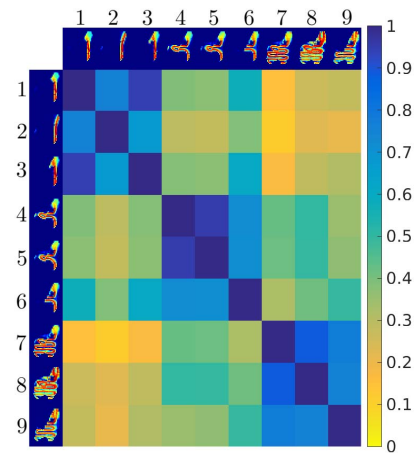

Fig. 19.   NSI affinity matrix including maps with missing data. The characteristics of the descriptor and NSI allow for a smooth degradation of the affinity value when missing data.

capture with the descriptor. For these experiments, we empirically chose 5*min* maps, that showed to be a good trade-off between computational cost and data filtering. Another parameter is the period of time in which we search for clusters. In this paper, we use one day and two weeks but we can choose shorter or longer periods. Our methodology presents consistent results and similar clusters can be identified whether analyzing a day by itself or combined with other days, including smaller clusters.

Fig. 19 shows the affinity matrix for nine clusters in a period where one of the (seven) cameras broke. For each cluster, we show the mean of its maps. Clusters 7, 8 and 9 are similar except in the area where the camera failed, in the middle left side of 9. Similarly, cluster 6 corresponds to the same queue state as clusters 4 and 5. Our method degrades gracefully in the presence of perturbations and clusters 6 and 9 have high affinity with the corresponding configuration classes.

Although the methodology presented in this paper is unsupervised, when working with real data and operational scenarios, interaction with end users/experts is very useful. Because of the large amount of data, airport experts are not able to describe classes *a priori*. By providing only representative patterns, our methodology works as a filter and saves users from the overwhelming load of interpreting all data. When presented with the relevant patterns, the airport experts recognize the classes as meaningful and understand the affinity matrix. Finally, with these classes, and applying queueing and discrete events theory, we are able to compute individual and more accurate models for each state.

VIII. CONCLUSION

We proposed a new descriptor to characterize the flow of people in large public infrastructures. Based on 3D data and without tracking people, we provided an occupancy map of the space—a very useful indicator of the service performance. We proposed an unsupervised methodology to identify and cluster these occupancy maps, without knowing their number or shape/configuration *a priori*. The approach is divided into two main steps: identify regular and

irregular periods/maps, and cluster the regular maps into classes corresponding to the different queue configurations. This methodology proposes a continuous irregularity measure for each map. The first step of the approach uses this measure as a means to identify the most abnormal maps in the data set. The second step segments the regular maps, based on an estimate of the number of clusters, and then consolidates the clusters using the Normalized Subspace Inclusion (NSI) criterion.

The approach gave consistent results, independently of the quantity of data analyzed, with the descriptor and NSI having smooth deterioration in the presence of missing data.

The data originates from different days but it has a common pattern: the busiest hours occur in the same periods of the day. Finally, irregular maps exist mainly during late night, when maintenance operations are due.

As future work, we plan to augment the model to account for more information sources, *e.g.*, velocity, flight schedule, flight destiny, in order to strengthen the assessment of the service performance.

REFERENCES

[1] S. Denman, T. Kleinschmidt, D. Ryan, P. Barnes, S. Sridharan, and C. Fookes, "Automatic surveillance in transportation hubs: No longer just about catching the bad guy," *Expert Syst. Appl.*, vol. 42, no. 24, pp. 9449–9467, 2015.

[2] R. Felkel and D. Klann, "Comprehensive passenger flow management at Frankfurt Airport," *J. Airport Manage.*, vol. 6, no. 2, pp. 107–124, 2012.

[3] J. Vongkulbhisal, R. Cabral, J. P. Costeira, and F. De la Torre, "Motion from structure (MfS): Searching for 3D objects in cluttered point trajectories," in *Proc. IEEE CVPR*, Jun. 2016, pp. 5639–5647.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

10

IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS

[4] M. Soltanolkotabi and E. J. Candés, "A geometric analysis of subspace clustering with outliers," *Ann. Statist.*, vol. 40, no. 4, pp. 2195–2238, 2012.

[5] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

[6] W. Chen, F. Guo, and F.-Y. Wang, "A survey of traffic data visualization," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 2970–2984, Dec. 2015.

[7] P. P.-Y. Wu and K. Mengersen, "A review of models and model usage scenarios for an airport complex system," *Transp. Res. A, Policy Pract.*, vol. 47, pp. 124–140, Jan. 2013.

[8] G. Guizzi, T. Murino, and E. Romano, "A discrete event simulation to model passenger flow in the airport terminal," in *Proc. 11th WSEAS Int. Conf. Math. Methods Comput. Techn. Elect. Eng.*, vol. 2. 2009, pp. 427–434.

[9] S. L. Dorton and D. Liu, "Effects of baggage volume and alarm rate on airport security screening checkpoint efficiency using queuing networks and discrete event simulation," *Hum. Factors Ergonom. Manuf. Service Ind.*, vol. 26, no. 1, pp. 95–109, 2016.

[10] R. R. Gilliam, "An application of queueing theory to airport passenger security screening," *Interfaces*, vol. 9, no. 4, pp. 117–123, 1979.

[11] S. Takakuwa and T. Oyama, "Modeling people flow: Simulation analysis of international-departure passenger flows in an airport terminal," in *Proc. 35th Conf. Winter Simulation, Driving Innov., Winter Simulation Conf.*, 2003, pp. 1627–1634.

[12] R. de Lange, I. Samoilovich, and B. van der Rhee, "Virtual queuing at airport security lanes," *Eur. J. Oper. Res.*, vol. 225, no. 1, pp. 153–165, 2013.

[13] A. Albiol, A. Albiol, and J. Silla, "Statistical video analysis for crowds counting," in *Proc. IEEE ICIP*, Nov. 2009, pp. 2569–2572.

[14] J. Barandiaran, B. Murguia, and F. Boto, "Real-time people counting using multiple lines," in *Proc. IEEE Image Anal. Multimedia Interact. Services (WIAMIS)*, May 2008, pp. 159–162.

[15] C. Gao, J. Liu, Q. Feng, and J. Lv, "People-flow counting in complex environments by combining depth and color information," *Multimedia Tools Appl.*, vol. 75, no. 15, pp. 9315–9331, 2016.

[16] L. Del Pizzo, P. Foggia, A. Greco, G. Percannella, and M. Vento, "A versatile and effective method for counting people on either RGB or depth overhead cameras," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jun./Jul. 2015, pp. 1–6.

[17] H. Fu, H. Ma, and H. Xiao, "Scene-adaptive accurate and fast vertical crowd counting via joint using depth and color information," *Multimedia Tools Appl.*, vol. 73, no. 1, pp. 273–289, 2014.

[18] Z. Zhao, H. Li, R. Zhao, and X. Wang, "Crossing-line crowd counting with two-phase deep neural networks," in *Proc. ECCV*, 2016, pp. 712–726.

[19] C. Zhang, H. Li, X. Wang, and X. Yang, "Cross-scene crowd counting via deep convolutional neural networks," in *Proc. CVPR*, Jun. 2015, pp. 833–841.

[20] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *Proc. IEEE CVPR*, Jun. 2016, pp. 589–597.

[21] Z. Ma and A. B. Chan, "Crossing the line: Crowd counting by integer programming with local features," in *Proc. IEEE CVPR*, Jun. 2013, pp. 2539–2546.

[22] S. Mukherjee, S. Gil, and N. Ray, "Unique people count from monocular videos," *Vis. Comput.*, vol. 31, no. 10, pp. 1405–1417, 2015.

[23] H. Nallaivarothayan, C. Fookes, S. Denman, and S. Sridharan, "An MRF based abnormal event detection approach using motion and appearance features," in *Proc. IEEE AVSS*, Aug. 2014, pp. 343–348.

[24] X. Wang, X. Ma, and W. E. L. Grimson, "Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 3, pp. 539–555, Mar. 2009.

[25] M. J. Roshtkhari and M. D. Levine, "Online dominant and anomalous behavior detection in videos," in *Proc. IEEE CVPR*, Jun. 2013, pp. 2611–2618.

[26] A. M. Cheriyadat and R. J. Radke, "Detecting dominant motions in dense crowds," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 4, pp. 568–581, Aug. 2008.

[27] J. Lei, T. Jiang, K. Wu, H. Du, G. Zhu, and Z. Wang, "Robust *K*-means algorithm with automatically splitting and merging clusters and its applications for surveillance data," *Multimedia Tools Appl.*, vol. 75, no. 19, pp. 12043–12059, 2016.

[28] M. M. Kalayeh, S. Mussmann, A. Petrakova, N. da Vitoria Lobo, and M. Shah. (2015). "Understanding trajectory behavior: A motion pattern approach." [Online]. Available: https://arxiv.org/abs/1501.00614

[29] B. T. Morris and M. M. Trivedi, "Learning, modeling, and classification of vehicle track patterns from live video," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 3, pp. 425–437, Sep. 2008.

[30] T. W. Liao, "Clustering of time series data—A survey," *Pattern Recognit.*, vol. 38, no. 11, pp. 1857–1874, 2005.

[31] Z. Hou and X. Li, "Repeatability and similarity of freeway traffic flow and long-term prediction under big data," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 6, pp. 1786–1796, Jun. 2016.

[32] Z. Wu and R. J. Radke, "Real-time airport security checkpoint surveillance using a camera network," in *Proc. CVPR Workshops*, 2011, pp. 25–32.

[33] J. Candamo, M. Shreve, D. B. Goldgof, D. B. Sapper, and R. Kasturi, "Understanding transit scenes: A survey on human behavior-recognition algorithms," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 1, pp. 206–224, Mar. 2010.

[34] J. Carvalho, M. Marques, J. P. Costeira, and P. M. Jorge, "Detecting people in large crowded spaces using 3D data from multiple cameras," in *Proc. 11th Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, 2016, pp. 218–225.

[35] N. P. da Silva and J. P. Costeira, "The normalized subspace inclusion: Robust clustering of motion subspaces," in *Proc. IEEE CVPR*, Sep./Oct. 2009, pp. 1444–1450.

[36] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. IT-28, no. 2, pp. 129–137, Mar. 1982.

[37] L. Kaufman and P. J. Rousseeuw, *Clustering by Means of Medoids*. Amsterdam, The Netherlands: North Holland, 1987.

[38] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining (KDD)*, Portland, Oregon, 1996.

João Carvalho received the M.Sc. degree in electrical and computer engineering from the Instituto Superior Técnico, Lisbon, Portugal, in 2012, where he is currently pursuing the Ph.D. degree under the Networked Cyber Physical Systems Doctoral Program. Since 2011, he has been a Research Assistant with the Institute for Systems and Robotics, Lisbon. From 2013 to 2014, he was a Research Assistant with Thales Portugal. His research interests include computer vision and robotics.

Manuel Marques received the Ph.D. degree in electrical and computer engineering from the Instituto Superior Técnico (IST) in 2011. He is currently a Researcher with the Instituto de Sistemas e Robótica, IST. He is a principal investigator of several projects, including the automatic survey of risk for cycling in cities using computer vision. His main areas of research include 3-D reconstruction, object recognition, video processing, and the applications to transportation and city mobility. He was a recipient of the Portuguese Science Foundation Fellowship.

João Paulo Costeira received the Ph.D. degree in electrical and computer engineering from the Instituto Superior Técnico (IST) in 1995. He was a Visiting Scientist with the Vision and Autonomous Systems Center, Carnegie Mellon University, from 1992 to 1995. He is currently an Associate Professor with the Electrical and Computer Engineering Department, IST. He is also a Researcher with the Instituto de Sistemas e Robótica, where he coordinates the Signal and Image Processing Group. He is also the Scientific Director of the CMU-Portugal Program, a joint venture between the Portuguese Government and Carnegie Mellon University. His research interests are in the area of 3D reconstruction from video, image matching, and recognition.