Integrated Analysis and Synthesis of Pedestrian Dynamics: First Results in a Real World Case Study

Sultan D. Khan, Luca Crociani, Giuseppe Vizzari
Complex Systems and Artificial Intelligence research center
Università degli Studi di Milano–Bicocca, Milano, Italy
{sultan.khan, luca.crociani, vizzari}@disco.unimib.it

***Abstract*—The paper introduces an agent-based model for the simulation of crowds of pedestrians whose main innovative element is the representation and management of an important type of social interaction among the pedestrians: members of groups, in fact, carry out of a form of interaction (by means of verbal or non-verbal communication) that allows them to preserve the cohesion of the group even in particular conditions, such as counter flows, presence of obstacles or narrow passages. The paper formally describes the model and presents its application to a real world scenario in which an analysis of the impact of groups on the overall observed system dynamics was performed. The simulation results are compared to empirical data and they show that the introduced model is able to produce quantitatively plausible results in situations characterised by the presence of groups of pedestrians.**

I. Introduction

The simulation of pedestrians and crowds is a consolidated and successful application of research results in the more general area of computer simulation of complex systems. Relevant contributions to this area come from disciplines ranging from physics and applied mathematics to computer science, often influenced by anthropological, psychological, sociological studies. The quality of the results provided by simulation models was sufficient to lead to the design and development of commercial software packages, offering useful functionalities to the end user (e.g. CAD integration, CAD-like functionalities, advanced visualisation and analysis tools) in addition to a simulation engine[1].

The last point is a crucial and critical element of this kind of research effort: computational models represent a way to formally and precisely define a computable form of theory of pedestrian and crowd dynamics. However, these theories must be validated employing field data, acquired by means of experiments and observations of the modeled phenomena, before the models can actually be used for sake of prediction. This paper represents a step in this direction, since it presents the application of methods from computer vision field for performing automated analysis on pedestrian dynamics, which are mainly aimed at the validation of an agent-based model for its simulation. The paper breaks down as the following. Description of the state-of-art of modelling and analysis of crowd dynamics is presented in Sec.II. Experimental methods for the automated analysis are described in Sec. III), while the real world case study used for their application is described in Sec. V-A. Then, the agent-based model for the simulation is presented in Sec. IV). First results of the analysis methods are described in Sec. V, while a discussion on the possibilities to exploit these data for sake of validation of the simulation results are presented in Sec. VI. Conclusions and future developments end the paper.

[1]See http://www.evacmod.net/?q=node/5 for a large list of pedestrian simulation models and tools.

II. Related Works

A. Synthesis

Pedestrian models can be roughly classified into three main categories that respectively consider pedestrians as *particles subject to forces*, particular *states of cells* in which the environment is subdivided in Cellular Automata (CA) approaches, or *autonomous agents* acting and interacting in an environment. The most widely adopted particle based approach is represented by the *social force model* [1], which implicitly employs fundamental proxemic concepts like the tendency of a pedestrian to stay away from other ones while moving towards his/her goal. *Cellular Automata* based approaches have also been successfully applied in this context: in particular, the floor-field model [2], in which the cells are endowed with a discretised gradient guiding pedestrians towards potential destinations. Finally, works like [3] essentially extend CA approaches, separating the pedestrians from the environment and granting them a behavioural specification that is generally more complex than what is generally represented in terms of a simple CA transition rule, but they essentially adopt similar methodologies. The resulting models are *agent–based*, since pedestrians are not merely states of cell. Along this direction of endowing models of more complicated behavioural models, relevant innovative studies regard social aspects and the transfer of emotions in crowds (see, e.g., [4]).

A recent survey of the field by [5] and by a report commissioned by the Cabinet Office by [6] made clear that, even after the substantial research that has been carried out in this area, there is still much room for innovations in models improving their performances both in terms of *effectiveness* in modelling pedestrians and crowd phenomena, in terms of *expressiveness* of the models (i.e. simplifying the modelling activity or introducing the possibility of representing phenomena that were still not considered by existing approaches), and in terms of *efficiency* of the simulation tools. Research on models able to represent and manage phenomena still not considered or properly managed is thus still lively and important. One of the aspects of crowds of pedestrians that has only been recently considered is represented by the implications of the presence of groups. A small number of recent works represent a relevant effort towards the modeling of groups, respectively in particle-based [7] (extending the social force model), in CA-based [8] (with ad-hoc approaches) and in agent-based approaches [9], [10] (introducing specific behavioral rules for managing group oriented behaviors): in all these approaches, groups are modeled by means of additional contributions to the overall pedestrian behaviour representing the tendency to stay close to other group members. However, the above approaches only mostly deal with small groups in relatively low density conditions; those dealing with relatively large groups (tens of pedestrians) were not validated against real data.

B. Automated Analysis

1) Dominant Flows Motion Detection: Crowd flow segmentation has multiple benefits: 1) enables clutter free visualization of moving groups 2) independence from detection and tracking 3) provide input for the pedestrian simulation models. Automatic analysis of the crowd has become the center of focus for most of researchers in computer vision. Detecting pedestrians and tracking are traditional ways of crowd analysis. Most algorithms developed for object detection and tracking work well in low density crowds where the number of people are less than twenty but in density crowds where the amount of people exceeds hundreds of thousand, detection and tracking of individuals are almost impossible due to multiple occlusions. Therefore, the research has focused on gathering global motion information at higher scale. Global analysis of dense group of moving people is often based on optical flow analysis.

A survey about the crowd analysis methods employed in computer vision is presented in [11]. An interdisciplinary framework for crowd analysis to improve simulation models of pedestrian flows is also presented in [12].

In [13] lagrangian coherent structures are detected by calculating finite-time scalar Lyapunov Exponent (FTLE) field over the phase space; these coherent structures represent different crowd motion patterns generating by moving in different directions. In [14] SIFT feature were instead used to detect dominant motion flows: flow vectors of SIFT features are calculated and then motion flow map is divided into small regions of equal size; in each region, dominant motion flows are estimated by clustering flow vectors. Crowd flow is estimated using multiple visual features reported in [15] where flow is estimated by the number of persons passing through a virtual trip wire and accumulate the total number of foreground pixels. Novel region growing scheme is adopted in [16] for crowd flow segmentation where translation flow is used to approximate the motion of crowd and region growing scheme is employed to segment the crowd flow. Min-cut/max flow algorithm is used in [17] for crowd flow segmentation. Histogram based crowd flow segmentation is reported in [18] where angle matrix of foreground pixels is segmented instead of optical flow foreground. The derivative curve of histogram is used to segment the flow.

2) People Counting in High Density Crowds: Estimating Crowd density and counting people is an important factor in crowd management. The increase of number of people in small areas may create problems like physical injury and fatalities. Hence early detection of the crowd can avoid these problems. Counting of the people moving in the crowd can provide information about the blockage at some point or even stampede. [19] proposed Bayesian model based segmentation to segment and count people but this method is not appropriate for high density crowds. [20] proposed blob features of moving objects to eliminate background and shadow from the image. [21] showed classification accuracy of 95% when crowd density is classified into four classes by using wavelet descriptors. [22] used texture descriptors called advanced local binary pattern descriptors to estimate crowd density estimation. [23] proposed a system that calculate the directional movement of the crowd and count the people as they cross some virtual line. [24] used specialized imaging system using infra-red imaging to count the people in the crowd. [25] have discussed in detail the concept of crowd monitoring using image processing through visual cameras. [26] used simple background subtraction from the static images to estimate the crowd density. Some other researchers [27], [28], [29] have also used the concept of background removal to estimate the crowd area. To estimate the crowd density using image processing, many researchers have used the information of texture, edges or some global or local features [30], [31].

III. Experimental Method for Automated Analysis

A. Motion Flow Characterisation

In this paper, we use Horn & Schunck [32] to calculate the dense optical flow field. The dense optical field calculated also contains the background information. We remove the background optical flow vectors by setting up a threshold. The optical flow vectors that are coherent and are a part of same flow are clustered. After clustering, some blobs appear which are removed by blob absorption method.

1) Motion Flow Composition: The motion flow field is a set of independent flow vectors in each frame and each flow vector is associated with its respective spatial location. The motion flow field is calculated by using optical flow methods. Given two images, F_t and F_{t+1} as input, we use Horn and Schunck [32] to compute dense optical flow. Consider a feature point i in F_t, its flow vector Z_i includes its location $X_i = (x_i, y_i)$ and its velocity $V_i = (v_{x_i}, v_{y_i})$, i. e. $Z_i = (X_i, V_i)$. We denote by $R_i(Z_i)$ as the magnitude of a flow vector and θ_i its angle or direction. The vector $M_i = (Z_i, R_i, \theta_i)$ summarises all the information associated to a feature i. Then $\{M_1, M_2, \ldots, M_k\}$ is the motion flow field of all the points of an image comprising $r \times c$ features such that $r \times c = k$, with r the number of rows and c the number of columns.

When computing dense optical flow we calculate the movement of all the pixels of an image, so it is usually the best

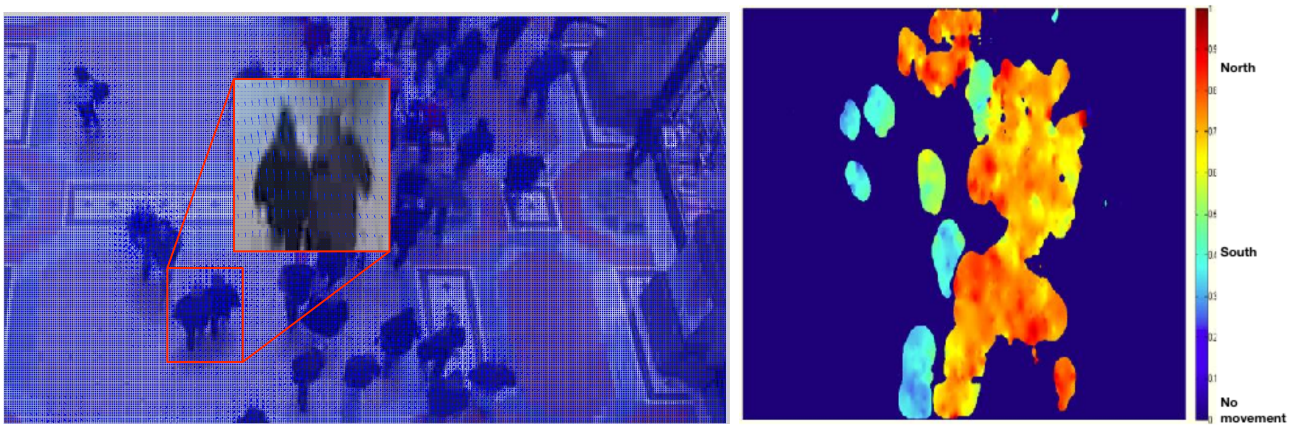

Fig. 1. Optical flow computed using the method described in [32], on the left, and compact representation of the movement direction of all pixels, on the right.

choice to remove background, to reduce computational costs without losing much information. To do so, a magnitude threshold is set to eliminate features characterised by a low magnitude that are considered background noise and that are not taken into account. This technique can also be used in computing coarse optical flow.

2) Motion Flow Field Segmentation: The motion flow field $\{M_1, M_2, \ldots, M_n\}$ is a matrix where each flow vector represents motion in specific direction as shown in Figure 1. Figure 1 does not show dominant motion patterns, so we can not infer any meaningful information about flow. Therefore, we need a method that automatically analyses the similarity among the flow vectors. We compute similarity among the flow vectors by applying similarity measure approach [33]. Similar vectors are grouped together to represent specific motion pattern by using clustering techniques [34]. This process of grouping vectors that represent specific motion pattern is called segmentation. After segmentation process, motion field is divided into small segments. Flow vectors that have similarity among them are clustered.

These blobs represent small clusters and resulted due to following reasons. First, if the objects move slowly, the inside and outside flow vectors of the objects are not same and as a result are classified into two different flows. Second, if the two opposite optical flow intersect, the optical flow at intersection point is ambiguous. Usually these small clusters or blobs are not the part of dominant motion flows. We adopt blob absorption approach, where these blobs are either absorbed by dominant cluster or by background. Let blob B_i of color C_i is the blob to be absorbed. Then find the edges of the blob by using Canny et al. [33] edge detector. For any point $p(x, y)$ on the edge of blob, we search 2×2 neighborhood of edge point. If any of neighborhood points has different color C_j that represents the dominant motion or background, then change the color of blob B_i to C_j, the color that represents dominant flow or background.

B. People Counting in a High Density Situations

In this paper, we have proposed a framework to count people in the extremely dense crowd where people are moving at different speeds. Foreground segmentation is done by various methods of background subtraction namely, approximate median, and frame difference and mixture of Gaussian method. Time complexity is calculated for these techniques and approximate median technique is selected which fast and accurate. Blob analysis is done to count the people in the crowd and blob area is optimized to get the best counting accuracy. In this paper, we extract the foreground by using Gaussian mixture model and optical flow. After getting foreground objects we use blob analysis method and optimize the blobs area by comparing it with ground truth data. Experimental results shows 90% accuracy of our results.

1) Motion Segmentation: Motion segmentation is the most important pre-processing step for detecting the moving objects from the video. Traditionally in video surveillance with a fixed camera, researchers tend to find some sort of motion in the video. There are two part of such of videos, background and foreground part. The object in motion is the foreground part of the video and the rest static part is the background. Motion detection is used to extract foreground part from the video. Such kind of extraction is useful for detecting, tracking and understanding the behavior of the object. Traditionally, background subtraction method is used for extracting moving objects from the video frame where pixels in the currents frame that deviate significantly from the background are considered as part of moving objects. Such kinds of methods are usually prone to errors due to unpredicted and changing behavior of the pixels. In addition, this method cannot accurately detect fast moving or slow moving as well as multiple objects. Also these methods are affected by change in illumination in the video frame. Sometime change in illumination in static background will be detected as part of moving object. Such errors and noise must be removed from the foreground objects before applying blob analysis. In order to extract valid and accurate foreground objects, we employed both Gaussian mixture model and Horn and Schank optical flow[32].

2) Blob Area Optimisation: Blobs are the connected regions in a binary image. For blob detection, image is first converted to binary image. Then next step is finding the connected components in the binary image. After finding connected components in binary image, the next step is to measure the properties of each connected component (object) in a binary image. In this paper, we are interested in measuring the 'Area' of each connected components. Area is the number of pixels in the region. Each binary image has a lot of connected components of variable size. We are interested in finding those connected components having area greater than some specific value. Area of the connected component differs depending upon the distance of camera from the scene. If the distance between the camera and crowd is less, greater will be number of pixels in a connected component and hence greater will be the blob size of the object. Hence the first step in people counting is to decide the optimal area of connected component. For this purpose, we have used four initial frames whose ground truth is available. In the iterative approach, we change the area of the blob size and count the people. This count is then compared with the ground truth of the frame (actual number of people in the frame). For each frame, optimal area is found for which the people count error was minimum.

IV. Pedestrian Simulation Model

In this section the formalisation of the agent-based computational model will be discussed, by focusing on the definition of its three main elements: *environment*, *update mechanism* and *pedestrian behaviour*.

A. Environment

The environment is modelled in a discrete way by representing it as a grid of squared cells with $40\ cm^2$ size (according to the average area occupied by a pedestrian [35]). Cells have a state indicating the fact that they are vacant or occupied by obstacles or pedestrians: $State(c) : Cells \rightarrow \{Free,\ Obstacle,\ OnePed_i, TwoPeds_{ij}\}$.

The last two elements of the definition point out if the cell is occupied by one or two pedestrians respectively, with their own identifier: the second case is allowed only in a controlled way to simulate overcrowded situations, in which the density is higher than $6.25\ m^{-2}$ (i.e. the maximum density reachable by our discretisation).

The information related to the scenario[2] of the simulation are represented by means of *spatial markers*, special sets of cells that describe relevant elements in the environment. In particular, three kinds of spatial markers are defined: (i) $start$ areas, that indicate the generation points of agents in the scenario. Agent generation can occur in *block*, all at once, or according to a user defined *frequency*, along with information on type of agent to be generated and its destination and group membership; (ii) $destination$ areas, which define the possible targets of the pedestrians in the environment; (iii) $obstacles$, that identify all the non-walkable areas as walls and zones where pedestrians can not enter.

Space annotation allows the definition of virtual grids of the environment, as containers of information for agents and their movement. In our model, we adopt the *floor field* approach [2], that is based on the generation of a set of superimposed grids (similar to the grid of the environment) starting from the information derived from spatial markers. Floor field values are spread on the grid as a gradient and they are used to support pedestrians in the navigation of the environment, representing their interactions with static object (i.e., destination areas and obstacles) or with other pedestrians. Moreover, floor fields can be *static* (created at the beginning and not changed during the simulation) or *dynamic* (updated during the simulation). Three kinds of floor fields are defined in our model: (i) *path field*, that indicates for every cell the distance from one destination area, acting as a potential field that drives pedestrians towards it (static). One path field for each destination point is generated in each scenario; (ii) *obstacles field*, that indicates for every cell the distance from neighbour obstacles or walls (static). Only one obstacles field is generated in each simulation scenario; (iii) *density field*, that indicates for each cell the pedestrian density in the surroundings at the current time-step (dynamic). Like the previous one, the density field is unique for each scenario.

Chessboard metric with $\sqrt{2}$ variation over corners [36] is used to produce the spreading of the information in the path and obstacle fields. Moreover, pedestrians cause a modification to the density field by adding a value $v = \frac{1}{d^2}$ to cells whose distance d from their current position is below a given threshold. Agents are able to perceive floor fields values in their neighbourhood by means of a function $Val(f, c)$ (f represents the field type and c is the perceived cell). This approach to the definition of the objective part of the perception model moves the burden of its management from agents to the environment, which would need to monitor agents anyway in order to produce some of the simulation results.

B. Pedestrians and Movement

Formally, our agents are defined by the following triple: $Ped = \langle Id,\ Group,\ State\rangle$; where $State = \langle position, oldDir,\ Dest\rangle$, with their own numerical identifier, their group (if any) and their internal state, that defines the current position of the agent, the previous movement and the final destination, associated to the relative path field.

Before describing agent behavioural specification, it is necessary to introduce the formal representation of the nature and structure of the groups they can belong to, since this is an influential factor for movement decisions.

1) Social Interactions: To represent different types of relationships, two kinds of groups have been defined in the model: a *simple group* indicates a family or a restricted set of friends, or any other small assembly of persons in which there is a strong and simply recognisable cohesion; a *structured group*

[2] It represents both the structure of the environment and all the information required for the realization of a specific simulation, such as crowd management demands (pedestrians generation profile, origin-destination matrices) and spatial constraints.

is generally a large one (e.g. team supporters or tourists in an organised tour), that shows a slight cohesion and a natural fragmentation into subgroups, sometimes simple.

Between members of a simple group it is possible to identify an apparent tendency to stay close, in order to guarantee the possibility to perform interactions by means of verbal or non-verbal communication [37]. On the contrary, in large groups people are mostly linked by the sharing of a common goal, and the overall group tends to maintain only a weak compactness, with a following behaviour between members. In order to model these two typologies, the formal representation of a group is described by the following: $Group : \langle Id, [SubGroup_1, \ldots, SubGroup_m], [Ped_1, \cdots, Ped_n] \rangle$.

In particular, if the group is simple, it will have an empty set of subgroups, otherwise it will not contain any direct references to pedestrians inside it, which will be stored in the respective leafs of its three structure. Differences on the modelled behavioural mechanism in simple/structured groups will be analysed in the following section, with the description of the utility function.

2) Agent Behaviour: Agent behaviour in a single simulation turn is organised into four steps: *perception, utility calculation, action choice* and *movement*. The *perception* step provides to the agent all the information needed for choosing its destination cell. In particular, if an agent does not belong to a group (from here called *individual*), in this phase it will only extract values from the floor fields, while in the other case it will perceive also the positions of the other group members within a configurable distance, for the calculation of the *cohesion* parameter. The choice of each action is based on an utility value assigned to every possible movement according to the function $U(c) = \frac{\kappa_g G(c) + \kappa_{ob} Ob(c) + \kappa_s S(c) + \kappa_c C(c) + \kappa_i I(c) + \kappa_d D(c) + \kappa_{ov} Ov(c)}{d}$.

Function $U(c)$ takes into account the behavioural components considered relevant for pedestrian movement, each one is modelled by means of a function that returns values in range $[-1; +1]$, if it represents an *attractive* element (i.e. its goal), or in range $[-1; 0]$, if it represents a *repulsive* one for the agent. For each function a κ coefficient has been introduced for its calibration: these coefficients, being also able to actually modulate tendencies based on objective information about agent's spatial context, complement the objective part of the perception model allowing agent heterogeneity. The purpose of the function denominator d is to constrain the diagonal movements, in which the agents cover a greater distance ($0.4 * \sqrt{2}$ instead of 0.4) and assume higher speed with respect to the non-diagonal ones.

The first three functions exploit information derived by local floor fields: $G(c)$ is associated to goal attraction whereas $Ob(c)$ and $S(c)$ respectively to geometric and social repulsion. Functions $C(c)$ and $I(c)$ are linear combinations of the perceived positions of members of agent group (respectively simple and structured) in an extended neighbourhood; they compute the level of attractiveness of each neighbour cell, relating to group cohesion phenomenon. Finally, $D(c)$ adds a bonus to the utility of the cell next to the agent according to his/her previous direction (a sort of *inertia* factor), while $Ov(c)$ describes the *overlapping* mechanism, a method used to allow two pedestrians to temporarily occupy the same cell at the same step, to manage high-density situations.

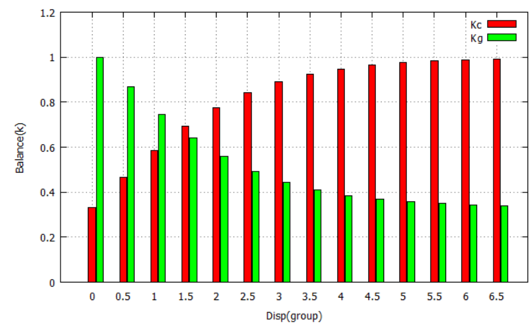

Fig. 2. Graphical representation of $Balance(k)$, for $k = 1$ and $\delta = 2.5$.

As we previously said, the main difference between simple and structured groups resides in the cohesion intensity, which in the simple ones is significantly stronger. Functions $C(c)$ and $I(c)$ have been defined to correctly model this difference. Nonetheless, various preliminary tests on benchmark scenarios show us that, used singularly, function $C(c)$ is not able to reproduce realistic simulations. Human behaviour is, in fact, very complex and can react differently even in simple situation, for example by allowing temporary fragmentation of simple groups in front of several constraints (obstacles or opposite flows). Acting statically on the calibration weight, it is not possible to achieve this dynamic behaviour: with a small cohesion parameter several permanent fragmentations have been reproduced, while with an increase of it we obtained no group dispersions, but also an excessive and unrealistic compactness.

In order to face this issue, another function has been introduced in the model, to adaptively balance the calibration weight of the three attractive behavioural elements, depending on the fragmentation level of simple groups:

$$Balance(k) = \begin{cases} \frac{1}{3} \cdot k + (\frac{2}{3} \cdot k \cdot DispBalance) & \textit{if } k = k_c \\ \frac{1}{3} \cdot k + (\frac{2}{3} \cdot k \cdot (1 - DispBalance)) & \textit{if } k = k_g \vee k = k_i \\ k & \textit{otherwise} \end{cases}$$

where $DispBalance = tanh(\frac{Disp(Group)}{\delta})$, $Disp(Group) = \frac{Area(Group)}{|Group|}$, k_i, k_g and k_c are the weighted parameters of $U(c)$, δ is the calibration parameter of this mechanism and $Area(Group)$ calculates the area of the convex hull defined using positions of the group members. Fig. 2 exemplifies both the group dispersion computation and the effects of the $Balance$ function on parameters. The effective utility computation, therefore, employs calibration weights resulting from this computation, that allows achieving a dynamic and adaptive behaviour of groups: cohesion relaxes if members are sufficiently close to each other and it intensifies with the growth of dispersion.

After the utility evaluation for all the cells in the neighbourhood, the choice of action is stochastic, with the probability to move in each cell c as (N is the normalization factor): $P(c) = N \cdot e^{U(c)}$. On the basis of $P(c)$, agents move in the resulted cell according to their set of possible actions, defined as list of the eight possible movements in the Moore neighbourhood, plus the action to keep the position (indicated as X): $A = \{NW, N, NE, W, X, E, SW, S, SE\}$.

C. Time and Update Mechanism

Time is also discrete: an initial definition of the duration of a time step was set to $0.31\ s$. This choice, considering the side of the cell (40 cm), generates a linear pedestrian speed of about $1.3\ m/s$, which is in line with the data from the literature representing observations of crowd in normal conditions [35].

Regarding the update mechanism, three different strategies are usually considered in this context [38]: *ordered sequential*, *shuffled sequential* and *parallel* update. The first two strategies are based on a sequential update of agents, respectively managed according either to a *static* list of priorities that reflects their order of generation or a *dynamic* one, shuffled at each time step. On the contrary, the parallel update calculates the choice of movement of all the pedestrians at the same time, actuating choices and managing collisions in a latter stage. In the model we adopted the parallel update strategy, that is usually considered more realistic due to consideration of conflicts arisen for the movement in a shared space [39], [40].

With this update strategy, the agents life-cycle must consider that, before carrying out the *movement* execution, potential conflicts[3] must be solved. The overall simulation step therefore follows a three step procedure: (i) *update of choices* and *conflicts detection* for each agent; (ii) *conflicts resolution*, that is the resolution of the detected conflicts between agent intentions; (iii) *agents movement*, that is the update of agent positions exploiting the previous conflicts resolution, and *field update*, that is the computation of the new density field according to the updated positions of the agents.

The resolution of conflicts employs an approach essentially based on the one introduced in [40], based on the notion of friction. Let us first consider that conflicts can involve two or more pedestrians: in case more than two pedestrians involved in a conflict for the same cell, the first step is to block all but two of them, randomly chosen, reducing the problem to a simple case. To manage a simple conflict, another random number $\in [0; 1]$is generated and compared to two thresholds, $frict_l$ and $frict_h$, with $0 < frict_l < frict_h \leq 1$: the outcome can be that all agents are blocked when the extracted number is lower than $frict_l$, only one agent moves (chosen randomly) when the extracted number is between $frict_l$ and $frict_h$ included, or even two agents move when the number is higher than $frict_h$ (in this case pedestrian overlapping occurs). For our tests, the values of the thresholds make it quite relatively unlikely the resolution of a simple conflict with one agent moving and the other blocked, and much less likely their overlapping.

[3]essentially related to the simultaneous choice of two (or more) pedestrians to occupy the same cell

Fig. 4. From the left: an overview of the Vittorio Emanuele II gallery and the quasi-zenithal of passerby within the walkway.

V. Experimental Results

This section discuss about the qualitative analysis of the results obtained from experiments. We carried out our experiments on a PC of 2.6 GHz (Core i5) with 4.0 GB memory and video by Bandini et al [41], whose scenario is described with the following.

A. The Analysed Scenario

The survey was performed the last 24$^{\text{th}}$ of November 2012 from about 2:50 pm to 4:10 pm. It consisted in the observation of the bidirectional pedestrian flows within the Vittorio Emanuele II gallery (see Fig. 4), a popular commercial-touristic walkway situated in the Milan city centre (Italy). The gallery was chosen as a crowded urban scenario, given the large amount of people that pass through it during the weekend for shopping, entertainment and visiting touristic-historical attractions in the centre of Milan.

The team performing the observation was composed of four people. Several preliminary inspections were performed to check the topographical features of the walkway. The balcony of the gallery, that surrounds the inside volume of the architecture from about ten meters in height, was chosen as location thanks to possibility to (i) position the equipment for video footages from a quasi-zenithal point of view and (ii) to avoid as much as possible to influence the behaviour of observed subjects, thanks to a railing of the balcony partly hiding the observation equipment. The equipment consisted of two professional full HD video cameras with tripods. The existing legislation about privacy was consulted and complied in order to comply with ethical issues about the privacy of people recorded within the pedestrian flows.

B. Automated Analysis Experimental Results

We computed optical flow at a coarser resolution to reduce computational time as shown in Figure 1(a); let the output of optical flow be a binary image F_b. The gaussian mixture model was applied to the same sample frame to compute foreground; let the output of the Gaussian mixture model be called F_{gm}. Later on, F_{gm} and F_b were put into logical AND

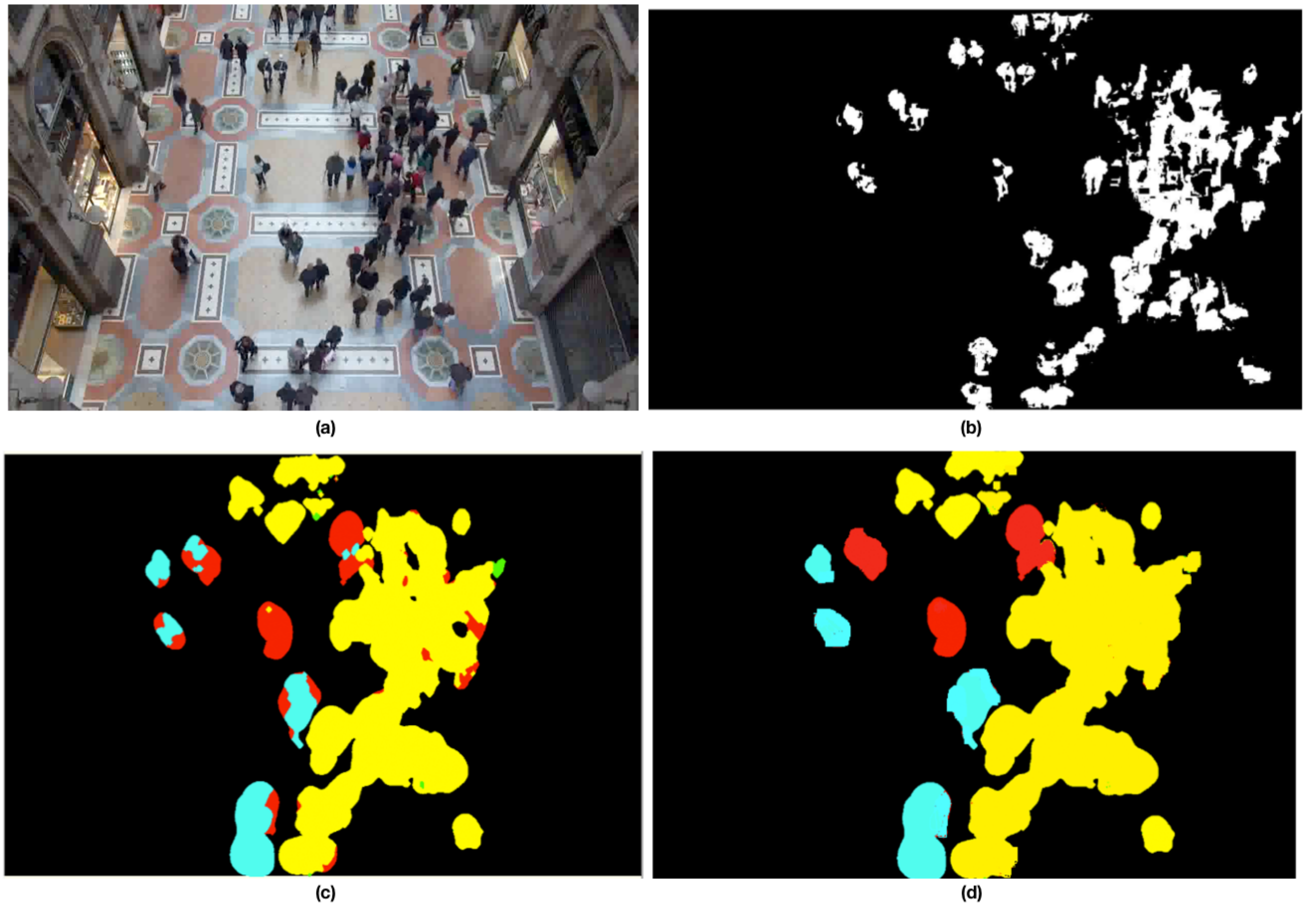

Fig. 3. Sample frame (a); its foreground image (F_f), computed by GMM and optical flow (b); the result of clustering flow vectors (c) and the result after applying blob absorption (d).

to extract foreground image F_f as shown in Figure 3(b). After computing optical flow and generating foreground image, similarity among the flow vectors in F_b was determined by using a similarity measure. Similar flow vectors were clustered to represent a specific motion pattern. Blob absorption method was applied to remove small clusters. Blob analysis method was applied on foreground image (F_f) to count the number of people.

Figure 3(c) and (d) show clustering result, flow vectors which satisfy similarity measures are combined into one cluster. Each cluster in the figure is color coded representing different motion patterns. Some blobs appear due to problems discussed in section III-A which are removed by the blob absorption method. Figure 3(d) shows a more refined version of motion patterns. As we can see from the Figure 3(c) and (d), there is a dominant motion towards North, a minor but still significant motion towards South, little motion towards West and almost negligible motion towards East.

Such little motion makes small clusters, which are usually absorbed by the blob absorption method, as shown in Figure 3(d). The results shows that large number of people moving towards North while there is less movement in other directions which is further illustrated in Figure 5.

Figure 5(a) shows ground truth calculated manually by using the Ground Truth Annotation (GTA) tool: 62 people were manually counted. After blob analysis, instead, 52 people were automatically detected in the whole scene as shown in Figure 5(b). Figure 5(c) shows instead the count of people moving in different directions: while studying dominant direction of crowd, analysis of speeds of crowd is important to understand overall crowd dynamics. Figure 5(d) shows the speed magnitude of all flow vectors. We use color codes to represent the magnitude of speeds. The bar scale represents different speeds where dark color or magnitude of 1 represent high speed while blue region shows zero magnitude. Figure 5(d) leads us to an important observation that the people moving alone are moving with high speed while others moving in groups move relatively slower. This kind of observations is important and provides a useful input to pedestrian simulation models.

Our proposed algorithm detects dominant motion patterns in the scene. Once motion pattern are detected, we can find the source and sink of each pattern. Sources refer to locations where objects appear and sinks are the location where objects disappear. Most of the scenes contain multiple sources and sinks: for example, a market place where multiple groups of

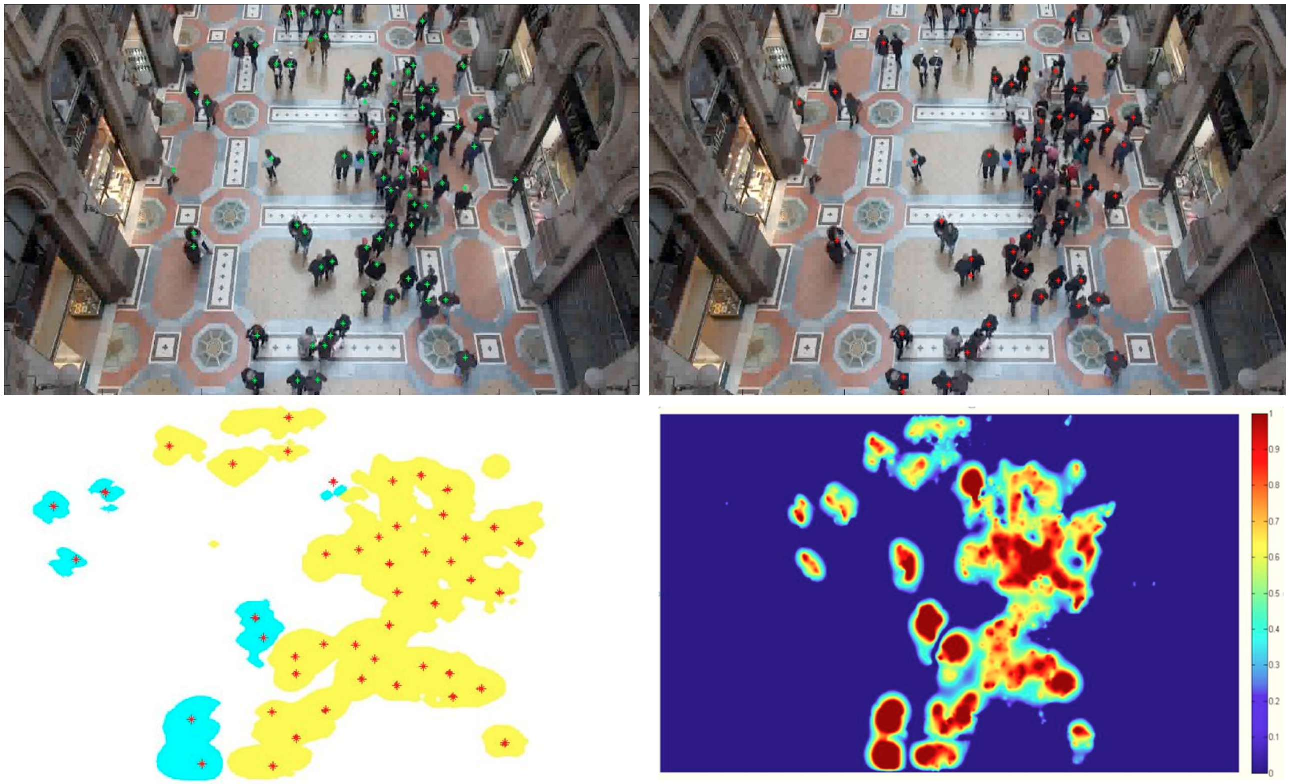

Fig. 5. The ground truth counting performed with GTA tool (a), people detected automatically by our algorithm (b); count of people in dominant flows (c) and the heat map describing speeds of pedestrians (d).

pedestrians move in distinct directions, originating multiple sources and sinks; similarly we could analyse flows in train stations or large floors of malls. The analysed video considers a situation in which, however, the flow of pedestrians is mostly on the North-South axis (referring to the video orientation). By analyzing sources and sinks of multiple motion patterns we can achieve information about the mostly visited or most attractive areas in the scene that can help us in understanding the behaviour of different pedestrian groups. In a transportation scenario, we could go as far as producing a so called origin–destination matrix which is an essential input for the creation of simulation scenarios. So, a generalisation of the presented work on dominant flows is object of current and future works.

VI. Discussion

The above described results of automated video analysis represent a first step in the direction of a more comprehensively integrated overall study of pedestrians and crowd dynamics. As of this moment, they still require some improvement to be directly applied, but in this section we will clarify some of the most immediate ways to exploit these results to support modeling and simulation.

A first employment of the above results is related to the actual configuration of the simulation scenario: qualitative analyses characterising the flow direction segmentation clarify that in the analysed portion of the environment most pedestrian movements are along the North-South axis (i.e. some pedestrians actually stop by a windows in one of the borders of the scenario or actually enter a shop, but their number is very low compared to the overall pedestrian flow). So, when designing the simulation environment we can exclude the presence of points of interest / attraction along the Eastern and Western borders of the environment: this was not obvious, since the analysed scenario comprises shops along the borders and since pedestrian behaviour in other areas of the gallery are quite different.

In particular, based on these data, we configured the environment as a large corridor with size 12.8 m $\times$ 13.6 m. At each end, one *start area* is placed for the agents generation, respecting the frequency of arrival observed in the videos; corridor ends also comprise a destination area corresponding to the start area positioned on the other end.

A second way to exploit data resulting from automated video analysis is represented by pedestrian counting and density estimation: the indication of the average number of pedestrians present in the simulated portion of the environment is actually important in configuring the start areas. In particular, in order to reproduce the counted number of pedestrians, also characterised according to their direction, we configured two different frequency profiles for the start areas which lead to achieve, respectively, 30 and 50 pedestrians in the environment on average. Should the automated analysis be able to discriminate different types of groups (i.e. individuals, couples, triples, etc.) this characterisation could

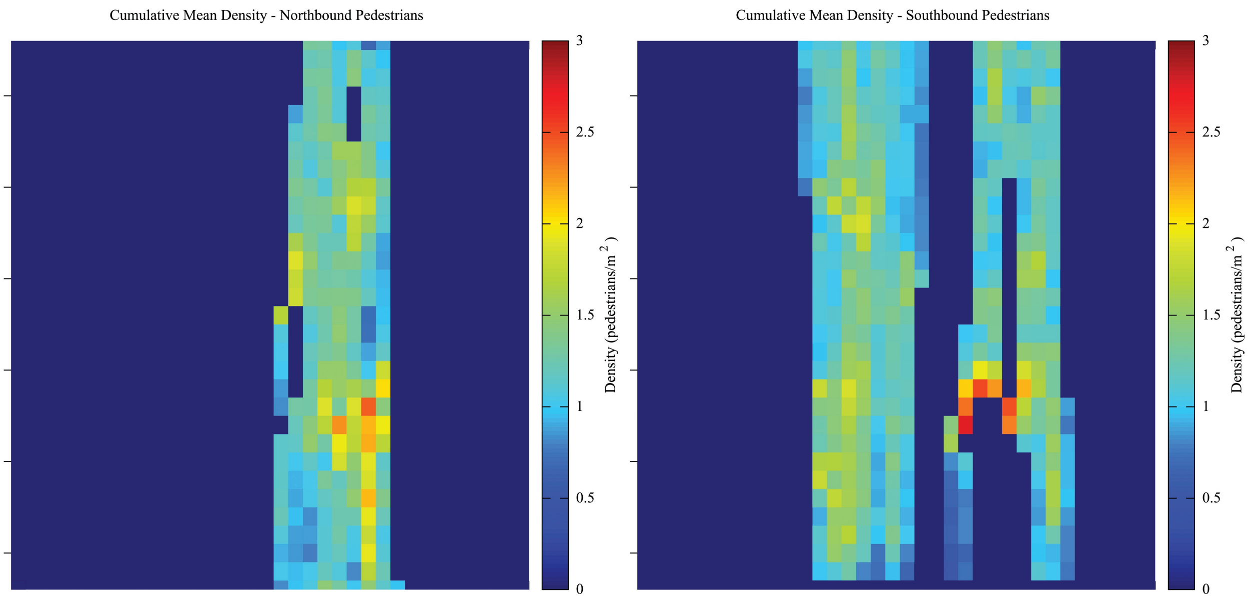

Fig. 6. Cumulative mean density maps calculated in a short period of the simulation (near 60 steps), which contains values of flow respectively towards North (a) and South (b).

further improve the starting areas configuration. The count of pedestrians in areas of the environment, and in particular portions of the overall analysed scene, could help generating Cumulative Mean Density maps [42], a heat-map diagram that can be used to validate simulation results. Examples of this type of result on the side of simulation (in the Galley scenario) are shown in Fig. 6, with an "instantaneous" (first 60 steps of the simulation, corresponding to 25 seconds of simulated time) calculation of the average *perceived*[4] local densities in the simulated environment, divided in the two directions of flow (i.e. North and South-bound). Even if it is only an example, this results is already able to provide useful information about the space utilization: while the flow towards North has remained relatively compact by forming one large lane, the one on the opposite direction has been divided in more lanes where little jamming are arisen, easily identifiable by the level of densities.

Finally, a third way to employ data resulting from automated video analysis is still related to the validation of simulation results and it still employs people counting data as a primary source. In particular, assuming that most of the counted pedestrians is actually in motion to enter or exit the monitored area (that could be a portion of the overall scene, like the northern border), we could estimated the instantaneous flow of pedestrians and average this value for certain time frames. The achieved measure can be compared to simulation results and this is particularly interesting in scenarios in which the density conditions have significant variations, since it could be possible to validate several points in the flow-density profile characterising the so called fundamental diagram [5] that can be achieved by means of simulation results.

[4]Values of local densities in each cell, contained in the *density* grid, are stored for the average calculation only when a pedestrian is located inside it.

Additional ways of employing other results of computer vision techniques for helping a simulation project, not necessarily discussed here or already employed for this analyses, could be done. For instance, once a source–sink analysis has been carried out, one could track a number of pedestrians completing a certain path and average out the travel time to have a reference value for evaluating simulations. However, the applicability and accuracy of tracking and many other techniques heavily depend on contextual factors like lightning conditions, changes in velocities and directions, but also crowding: a high density of pedestrians is very frequently always causing occlusions that can mislead the tracking algorithm. This work represents one first step in a more general research work aimed at contributing a fruitful interaction between pedestrian simulation and computer vision research producing (i) vertical results, namely techniques and case studies in specific contexts, and (ii) guidelines for the adoption of the most appropriate technique for a given context and situation.

VII. Conclusions

This paper has introduced the first results of a research effort putting together techniques of automated analysis of pedestrian and crowd dynamics and approaches towards the synthesis of these kind of phenomena. While in the present form the results of automated analysis mostly provide qualitative indications to the modeler, in the future they will represent also a quantitative empirical data for the initialization, calibration and validation of simulation models. The main contribution of the present work is represented by a systematic collaboration of automated analysis and simulation approaches in a specific and challenging real-world scenario, already producing useful indications that, however, will soon be improved for an even smoother and more quantitative integration. The main future directions are, on one hand, aimed at a more thorough quantification of the results of analyses and, on the other, at the identification and understanding the behaviour of pedestrian groups in the scene (e.g. source-sink analysis, converging to a point or dispersing, circling around points of reference).

REFERENCES

[1] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Phys. Rev. E*, vol. 51, no. 5, pp. 4282–4286, May 1995.

[2] C. Burstedde, K. Klauck, A. Schadschneider, and J. Zittartz, "Simulation of pedestrian dynamics using a two-dimensional cellular automaton," *Physica A: Statistical Mechanics and its Applications*, vol. 295, no. 3 - 4, pp. 507 – 525, 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378437101001418

[3] C. M. Henein and T. White, "Agent-based modelling of forces in crowds." in *Multi-Agent and Multi-Agent-Based Simulation, Joint Workshop MABS 2004, New York, NY, USA, July 19, 2004, Revised Selected Papers*, ser. Lecture Notes in Computer Science, P. Davidsson, B. Logan, and K. Takadama, Eds., vol. 3415. Springer–Verlag, 2005, pp. 173–184.

[4] T. Bosse, M. Hoogendoorn, M. C. A. Klein, J. Treur, C. N. van der Wal, and A. van Wissen, "Modelling collective decision making in groups and crowds: Integrating social contagion and interacting emotions, beliefs and intentions," *Autonomous Agents and Multi-Agent Systems*, vol. 27, no. 1, pp. 52–84, 2013.

[5] A. Schadschneider, W. Klingsch, H. Klüpfel, T. Kretz, C. Rogsch, and A. Seyfried, "Evacuation dynamics: Empirical results, modeling and applications," in *Encyclopedia of Complexity and Systems Science*, R. A. Meyers, Ed. Springer, 2009, pp. 3142–3176.

[6] R. Challenger, C. W. Clegg, and M. A. Robinson, "Understanding crowd behaviours: Supporting evidence," http://www.cabinetoffice.gov.uk/news/understanding-crowd-behaviours, University of Leeds, Tech. Rep., 2009.

[7] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz, "The walking behaviour of pedestrian social groups and its impact on crowd dynamics," *PLoS ONE*, vol. 5, no. 4, p. e10047, 04 2010. [Online]. Available: http://dx.doi.org/10.1371%2Fjournal.pone.0010047

[8] S. Sarmady, F. Haron, and A. Z. H. Talib, "Modeling groups of pedestrians in least effort crowd movements using cellular automata," in *Asia International Conference on Modelling and Simulation*, D. Al-Dabass, R. Triweko, S. Susanto, and A. Abraham, Eds. IEEE Computer Society, 2009, pp. 520–525.

[9] R. A. Rodrigues, A. de Lima Bicho, M. Paravisi, C. R. Jung, L. P. Magalhães, and S. R. Musse, "An interactive model for steering behaviors of groups of characters," *Applied Artificial Intelligence*, vol. 24, no. 6, pp. 594–616, 2010.

[10] J. Tsai, N. Fridman, E. Bowring, M. Brown, S. Epstein, G. A. Kaminka, S. Marsella, A. Ogden, I. Rika, A. Sheel, M. E. Taylor, X. Wang, A. Zilka, and M. Tambe, "Escapes - evacuation simulation with children, authorities, parents, emotions, and social comparison," in *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems – Innovative Applications Track (AAMAS 2011)*, Tumer, Yolum, Sonenberg, and Stone, Eds., 2011, pp. 457–464.

[11] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, and L.-Q. Xu, "Crowd analysis: a survey," *Mach. Vis. Appl.*, vol. 19, no. 5-6, pp. 345–357, 2008.

[12] M. Butenuth, F. Burkert, F. Schmidt, S. Hinz, D. Hartmann, A. Kneidl, A. Borrmann, and B. Sirmaçek, "Integrating pedestrian simulation, tracking and event detection for crowd analysis," in *ICCV Workshops*. IEEE, 2011, pp. 150–157.

[13] S. Ali and M. Shah, "A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis," in *CVPR*. IEEE Computer Society, 2007.

[14] O. Ozturk, T. Yamasaki, and K. Aizawa, "Detecting dominant motion flows in unstructured/structured crowd scenes," in *ICPR*. IEEE, 2010, pp. 3533–3536.

[15] S. Srivastava, K. K. Ng, and E. J. Delp, "Crowd flow estimation using multiple visual features for scenes with changing crowd densities," in *AVSS*. IEEE Computer Society, 2011, pp. 60–65.

[16] S. Wu, Z. Yu, and H.-S. Wong, "Crowd flow segmentation using a novel region growing scheme," in *PCM*, ser. Lecture Notes in Computer Science, P. Muneesawang, F. Wu, I. Kumazawa, A. Roeksabutr, M. Liao, and X. Tang, Eds., vol. 5879. Springer, 2009, pp. 898–907.

[17] H. Ullah and N. Conci, "Crowd motion segmentation and anomaly detection via multi-label optimization," in *ICPR workshop on Pattern Recognition and Crowd Analysis*, 2012.

[18] C. Shiyao, L. Nianqiang, and L. Zhen, "Multi-directional crowded objects segmentation based on optical flow histogram," in *Image and Signal Processing (CISP), 2011 4th International Congress on*, vol. 1. IEEE, 2011, pp. 552–555.

[19] T. Zhao and R. Nevatia, "Bayesian human segmentation in crowded situations," in *CVPR (2)*. IEEE Computer Society, 2003, pp. 459–466.

[20] S. Yoshinaga, A. Shimada, and R.-i. Taniguchi, "Real-time people counting using blob descriptor," *Procedia-Social and Behavioral Sciences*, vol. 2, no. 1, pp. 143–152, 2010.

[21] L. Xiaohua, S. Lansun, and L. Huanqin, "Estimation of crowd density based on wavelet and support vector machine," *Transactions of the Institute of Measurement and Control*, vol. 28, no. 3, pp. 299–308, 2006.

[22] W. Ma, L. Huang, and C. Liu, "Advanced local binary pattern descriptors for crowd estimation," in *PACIIA (2)*. IEEE Computer Society, 2008, pp. 958–962.

[23] D. Yoshida, K. Terada, S. Oe, and J.-I. Yamaguchi, "A method of counting the passing people by using the stereo images," in *ICIP (2)*, 1999, pp. 338–342.

[24] K. Hashimoto, K. Morinaka, N. Yoshiike, C. Kawaguchi, and S. Matsueda, "People count system using multi-sensing application," in *Solid State Sensors and Actuators, 1997. TRANSDUCERS'97 Chicago., 1997 International Conference on*, vol. 2. IEEE, 1997, pp. 1291–1294.

[25] A. Davies, J. H. Yin, and S. Velastin, "Crowd monitoring using image processing," *Electronics Communication Engineering Journal*, vol. 7, no. 1, pp. 37–47, 1995.

[26] D. Roqueiro and V. A. Petrushin, "Counting people using video cameras," *IJPEDS*, vol. 22, no. 3, pp. 193–209, 2007.

[27] S. Velastin, J. Yin, A. Davies, M. Vicencio-Silva, R. Allsop, and A. Penn, "Analysis of crowd movements and densities in built-up environments using image processing," in *Image Processing for Transport Applications, IEE Colloquium on*. IET, 1993, pp. 8–1.

[28] ——, "Automated measurement of crowd density and motion using image processing," in *Road Traffic Monitoring and Control, 1994., Seventh International Conference on*. IET, 1994, pp. 127–132.

[29] A. Marana, S. Velastin, L. Costa, and R. Lotufo, "Estimation of crowd density using image processing," in *Image Processing for Security Applications (Digest No.: 1997/074), IEE Colloquium on*. IET, 1997, pp. 11–1.

[30] R. Ma, L. Li, W. Huang, and Q. Tian, "On pixel count based crowd density estimation for visual surveillance," in *Cybernetics and Intelligent Systems, 2004 IEEE Conference on*, vol. 1. IEEE, 2004, pp. 170–173.

[31] S.-F. Lin, J.-Y. Chen, and H.-X. Chao, "Estimation of number of people in crowded scenes using perspective transformation," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 31, no. 6, pp. 645–654, 2001.

[32] B. K. P. Horn and B. G. Schunck, """determining optical flow": A retrospective," *Artif. Intell.*, vol. 59, no. 1-2, pp. 81–87, 1993.

[33] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, 1986.

[34] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, 1999.

[35] U. Weidmann, "Transporttechnik der fussgänger - transporttechnische eigenschaftendes fussgängerverkehrs (literaturstudie)," Institut füer Verkehrsplanung, Transporttechnik, Strassen- und Eisenbahnbau IVT an der ETH Zürich, Literature Research 90, 1993.

[36] T. Kretz, C. Bönisch, and P. Vortisch, "Comparison of various methods for the calculation of the distance potential field," in *Pedestrian and Evacuation Dynamics 2008*, W. W. F. Klingsch, C. Rogsch, A. Schadschneider, and M. Schreckenberg, Eds. Springer Berlin Heidelberg, 2010, pp. 335–346. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-04504-2_29

[37] M. Costa, "Interpersonal distances in group walking," *Journal of Nonverbal Behavior*, vol. 34, pp. 15–26, 2010, 10.1007/s10919-009-0077-y. [Online]. Available: http://dx.doi.org/10.1007/s10919-009-0077-y

[38] H. Klüpfel, "A cellular automaton model for crowd movement and egress simulation," Ph.D. dissertation, University Duisburg-Essen, 2003.

[39] A. Kirchner, A. Namazi, K. Nishinari, and A. Schadschneider, “Role of Conflicts in the Floor Field Cellular Automaton Model for Pedestrian Dynamics,” in *2nd International Conference on Pedestrian and Evacuation Dynamics*, 2003.

[40] A. Kirchner, K. Nishinari, and A. Schadschneider, “Friction effects and clogging in a cellular automaton model for pedestrian dynamics,” *Phys. Rev. E*, vol. 67, p. 056122, May 2003. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevE.67.056122

[41] S. Bandini, A. Gorrini, and G. Vizzari, “Towards an integrated approach to crowd analysis and crowd synthesis: a case study and first results,” *CoRR*, vol. abs/1303.5029, 2013.

[42] C. Castle, N. Waterson, E. Pellissier, and S. Bail, “A comparison of grid-based and continuous space pedestrian modelling software: Analysis of two uk train stations,” in *Pedestrian and Evacuation Dynamics*, R. D. Peacock, E. D. Kuligowski, and J. D. Averill, Eds. Springer US, 2011, pp. 433–446. [Online]. Available: http://dx.doi.org/10.1007/978-1-4419-9725-8_39