

Tracking People Motion Based on Extended Condensation Algorithm

Jorge García, Alfredo Gardel, Ignacio Bravo, José Luis Lázaro, and Miguel Martínez

Abstract—People counting systems are widely used in surveillance applications. In this paper, we present a solution to bidirectional people counting based on information provided by an overhead stereo system. Four fundamental aspects can be identified: the detection and tracking of human motion using an extended particle filter, the use of 3-D measurements in order to increase the system's robustness and a modified K-means algorithm to provide the number of hypotheses at each time, and, finally, trajectory generation to facilitate people counting in different directions. The proposed algorithm is designed to solve problems of occlusion, without counting objects such as shopping trolleys or bags. A processing ratio of around 30 frames/s is necessary in order to capture the real-time trajectory of people and obtain robust tracking results. We validated various test videos, achieving a hit rate between 95% and 99%, depending on the number of people crossing the counting area.

Index Terms—Extended condensation algorithm, motion detection, people counting, tracking people.

I. INTRODUCTION

MANY video surveillance applications exist in which people counting provides important information for different tasks since, nowadays, there are many public and private buildings where large groups of people are concentrated. In emergency situations, it is crucial to be able to locate these people in order for the appropriate authorities to organize evacuation.

Similarly, many businesses within the service sector need to conduct statistical analyses of their sales access points in order to monitor customer flow at different times of the day and adjust sales personnel presence accordingly, identify the areas which customers visit most, etc. Such systems are usually positioned at the entrance and exit doors of buildings, department store aisles, hospital corridors, etc.

As reported in [1], the problem of people counting has generally been addressed using relatively inefficient systems, such as turnstiles to reduce the flow of people moving through an area.

Manuscript received May 16, 2011; revised October 13, 2012 and January 24, 2012; accepted May 31, 2012. Date of publication February 1, 2013; date of current version April 12, 2013. This research was supported in part by the Spanish Research Program (Programa Nacional de Diseño y Producción Industrial, Ministerio de Ciencia y Tecnología) through the project ESPIRA (ref. DPI2009-10143) and in part by the University of Alcalá (ref. UAH2011/EXP-001) through the project "Sistema de Arrays de Cámaras Inteligentes." This paper was recommended by Associate Editor D. Zhang.

The authors are with the Department of Electronics, Escuela Politécnica Superior, University of Alcalá, 28871 Alcalá de Henares (Madrid), Spain (e-mail: jorge.garcia@depeca.uah.es; alfredo@depeca.uah.es; ibravo@depeca.uah.es; lazaro@depeca.uah.es; miguel.martinez@depeca.uah.es).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSMCA.2012.2220540

Other systems have been based on optical barriers, which produce a high error rate both in terms of false negatives, by failing to discriminate between different people walking in parallel, and false positives, by including objects such as bags and cases in the people count.

Several alternatives to these systems are based on computer vision. These constitute low-cost nonintrusive systems which are capable of resolving some of the problems mentioned earlier and which yield a relatively high hit rate. In addition, some of these systems have the ability to track people crossing the camera's field of view, increasing robustness by taking several measurements corresponding to the same person in the video sequence.

In this paper, we present a bidirectional people counting system based on computer vision and propose solutions to various common problems such as the occlusion of people and discriminating between people and objects such as shopping trolleys or bags in stores.

This paper is divided into the following sections. In Section II, we review previous studies related to the subject under discussion; in Section III, we present the proposed counting system; and, finally, in Section IV, we provide an analysis of the results.

II. RELATED WORK

Numerous studies in the literature have proposed solutions to people counting using computer vision [2], [3]. The most important factor to consider is the position and orientation of the camera(s) used for the counting system since different problems may emerge depending on how the scene is captured. In addition, different processing algorithms are required according to the position and orientation selected.

The most common location about camera systems has been at a specific height in relation to the floor, always overhead. However, in some studies, such as [4], the camera was positioned at head height since the system was based on face detection. As regards orientation, in some studies, the cameras were tilted, such as in [5], whereas in others, an overhead camera was used [6].

A priori, neither orientations is optimal, as each presents advantages and disadvantages. Tilted cameras can provide many details concerning the people moving through the counting area, such as the position of body parts, skin, the omega (Ω) silhouette formed by the head and shoulders, etc. However, occlusion can be a problem with such systems, as reported in [7]. In some studies, such as [8], occlusion problems were solved by increasing the number of cameras, all located in different positions. Thus, different projections of the scene were obtained, and the problem of occlusion was solved. Where an

overhead camera has been used, such as in [9] and in the present proposal, the problem of occlusion is solved completely, but the possibility of extracting detailed information about people is drastically reduced. Thus, the decision to use one approach or another will be heavily influenced by the type of people detection required. However, the system should be capable of differentiating people from other kinds of moving objects which may appear in the scene, and overhead cameras would, in principle, appear to be more effective than tilted cameras.

One method that several systems have used to detect moving targets has been to capture the foreground of the scene, discarding everything in the background, as in [10] and [11]. The subtraction of the background implies the inclusion of an updating algorithm such as the Gaussian mixture background model (*GMBM*), as proposed in [12]. This, in turn, implies an increase in execution time (t_{exe}), a critical parameter in real-time systems.

In this paper, in contrast, we propose the use of the consecutive temporal image difference method, which is a method for tracking movement in a scene that requires a minimum t_{exe} . However, one of the problems with this method is that it is only possible to track motion in the scene [13], and it thus becomes necessary to carry out some kind of postprocessing, such as generating blobs for each area containing movement, in order to extract the features required to carry out detection. Nevertheless, this method is capable of adapting to abrupt changes in illumination, whereas the previously mentioned system (*GMBM*) requires the time t_{update} , leading to erroneous results. In [14], an overhead camera is presented. Following the prior processing of the captured image, more meaningful blobs are obtained. In order to differentiate between people and objects, objects are defined as being likely to present straight lines in their outlines. Using the Hough transform technique on images of the blob edge, the system evaluates the image corresponding to a person or an object, obtaining 90% to 100% efficiency, depending on the number of people in the counting area.

In this paper, we propose the use of a more robust method for differentiating between people and the possible objects which may move through the counting area, obtaining the depth of the targets via a stereo system. It becomes relatively simple to differentiate between people and objects when depth is incorporated into a stereo system since people present much less depth than flat objects. Accordingly, by assessing depth in areas where motion has been detected, it is possible to detect the people present in the counting area. Our aim was to obtain a minimum efficiency rate of around 95%, considering this as the percentage necessary to validate the counting application.

Numerous studies have attempted to resolve the problem of multitarget tracking, as described in this paper. In order to carry out this task, three proposals have been put forward [15]:

- 1) the use of one estimator for each of the tracked targets;
- 2) the use of a single estimator, increasing the estimated state vector so that this includes the components of each of the tracked targets;
- 3) the use of a single multimodal estimator.

The first two proposals present the disadvantage of increasing the t_{exe} as the number of tracked targets increases. In the

first case, it is necessary to include as many estimators as tracked targets, and in the second case, it is necessary to add as many state components as targets. However, the third proposal maintains a practically constant t_{exe} regardless of the number of targets since it includes all of them in the *a priori* belief. It is for this reason that we decided to use a multimodal estimator since particle filtering presented the best probability estimation option for estimating nonlinear non-Gaussian dynamic processes.

In [16], the task of tracking targets was addressed through the use of sequential importance sampling (SIS), a more basic version of particle filtering. Meanwhile, in [17], sampling importance resampling (SIR) was combined with a genetic algorithm in order to make use of state-space modeling and evolutionary computation, respectively. Similarly, in [18], the SIR filter was combined with a detection algorithm to estimate 3-D position. This SIR filter resolved the problems of data set degeneration presented by the SIS filter. The bootstrap particle filter, a version of the SIR filter, was used in [19] to track blobs. None of the filters cited thus far are multimodal. In this paper, we propose using the extended condensation algorithm with a modification to the reinitialization step, yielding an extended particle filter with random reinitialization. This filter constitutes a modified version of the bootstrap algorithm, enabling it to adapt to multimode estimation tasks, improving the accuracy and hit rate of similar approaches such as that used in [20].

III. METHODOLOGY

We propose the use of the extended modified condensation algorithm, based on optical flow generated from the movement of people and depth to the height of the system, as an estimation method for multiple people. The modified K-means algorithm is used to provide a deterministic output. The inclusion of different features relevant to people tracking, such as movement, size, and height, adapting the propagation and observation models in the particle filter and followed by a clustering method, provides sufficient accuracy and robustness to achieve high counting rates.

Fig. 1 shows the different stages involved in the proposed system. First, stereo images are rectified (1.a), and then, image motion and people candidates' heights are calculated (1.b). This information is used to update the extended particle filter (1.c). Clusters are determined from the set of particles representing the different people candidates (1.d). Finally, people trajectories are generated in order to perform people counting (1.e).

A. Image Rectification

Epipolar geometry is an intrinsic projective geometry between two views and depends on the internal parameters of the cameras and their relative position [21]. This geometry is widely used to obtain the 3-D position of a point in space with respect to the camera system, by seeking corresponding points in paired stereo images. Epipoles are located at infinity in the configuration of the proposed system; thus, the optical axes are parallel, and the image planes are coplanar. The considerable advantage of this configuration is that, when searching for the correspondence of a point in the image plane, it is only necessary to search along a single line.

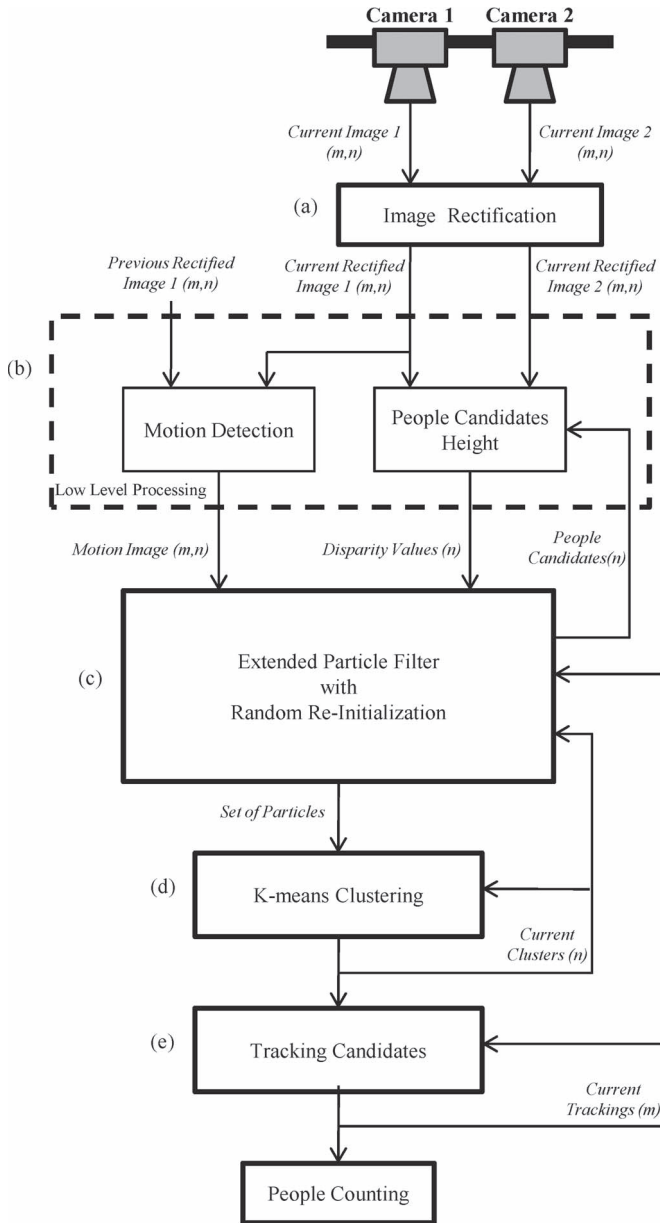


Fig. 1. Image rectification.

To obtain the same perspective of the space from both cameras, it is necessary to rectify the images provided by the stereo system, in order to comply with the constraints of the geometry described. To this end, the intrinsic parameters for each camera and the extrinsic parameters of the stereo system are obtained *offline* using the method proposed by Zhang [22], while the images are rectified using *Bouguet's algorithm*. It should be noted that these images constitute the proposed algorithm's data input.

B. Low-Level Processing

It is worth noting that low-level processing provides information for the subsequent stages; thus, the lack of robustness and the inclusion of noisy blobs (i.e., shadows, lighting changes, etc.) will be resolved in the particle filter algorithm, considering the stereo measurement.

1) *Motion Detection*: We used the images provided by camera 1 to detect human motion. However, it is irrelevant whether the images of camera 1 or camera 2 are used since motion detection is carried out in the area covered by both images, and thus, the images provided by one of them will be redundant for this purpose. By subtracting consecutive images, the position where movement has occurred is obtained. Consecutive image differencing is a simple technique for extracting movement from the background with a very low computational cost. This method of motion detection merely requires the current and previous images and thus adapts rapidly to any changes in illumination, in contrast to background subtraction methods where illumination changes have a greater impact. Furthermore, this method does not pose problems when new static elements are added to the background while the system is running.

The image differencing results are thresholded to eliminate false detections produced by low levels of noise in the image. Furthermore, in this way, all the image pixels representing foreground targets present the same value since interest is not focused on this value, which provides no information, but focused on the situation in the image. The image motion I_m is given by the following:

$$\begin{cases} I_c(m, n) - I_p(m, n) \geq \text{THR}_m \\ I_c(m, n) - I_p(m, n) < \text{THR}_m \end{cases} \quad (1)$$

where I_c is the current rectified image, I_p is the previous rectified image, and THR_m is the threshold value. This is constant for those situations where lighting changes are minimal, such as indoors. In other situations, such as outdoor areas, this value will be updated at run time.

The average walking speed of a person is about 1.5 m/s, but the motion threshold should be set to detect motion with a minimum average speed of about 1 m/s. Several types of movements, such as *stop-and-go*, are processed satisfactorily if there is sufficient motion between consecutive frames of movement. Here, lifetime is the parameter to consider when establishing the inclusion of the object in the background.

2) *People Candidate Height*: This part of the algorithm obtains the height of people candidates given by auxiliary clusters generated in the current iteration. Thus, edge detection followed by stereo matching processing is carried out in different regions of interest. Motion detection areas are not directly related. The cluster positions assume that the data are filtered and robust when seeking the correct stereo correspondence.

Edges are important features since they depict significant changes in local intensity, defining the object for later recognition, and are used to provide sufficient characteristics for the subsequent calculation of correspondences in order to obtain different depth values. Stereo correspondence methods can be divided into two groups: methods based on areas which use the intensity values of stereo images and those based on characteristics extracted from stereo images such as edges, corners, etc. These latter are more stable methods [23], and therefore, an exact match is not imperative in order to obtain the object's height in the desired range with a modeled Gaussian error of $\mu = 5$ cm and $\sigma = 10$ cm. In this paper, we used a Canny edge detector [24]. This detector presented the best performance and robustness, in addition to obtaining all the edges in terms of

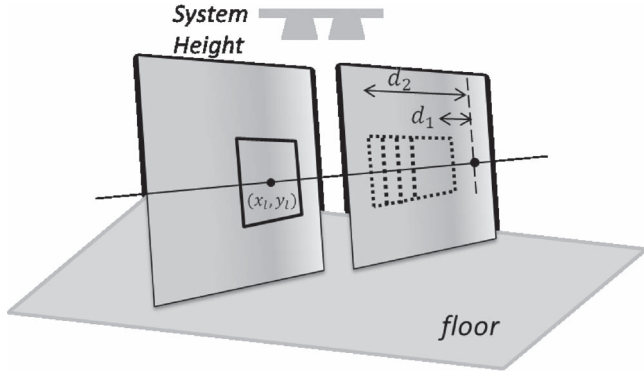


Fig. 2. Matching the possible disparity values.

direction as well as smoothing outlines. On the other hand, it has a higher running time than other types of detector. In the proposed algorithm, image edges are not computed for the whole image, but only in the region of interest *ROI* where there are people candidates. Thus, it is necessary to obtain stereo correspondences.

As reported in [25], the depth value in our specific configuration is calculated according to (2), as explained in the section on image rectification

$$\text{depth} = f \cdot \frac{B}{d_x} \quad (2)$$

where f is the focal distance common to the system, B is the distance between the cameras' optical centers (*baseline*), and d_x is the disparity. Focal distance and *baseline* are constant parameters established by the conditions of the stereo system and obtained by means of the stereoscopic calibration given in [22]. Disparity is calculated as the difference between coordinates on the x -axis of a point p_l with coordinates (x_l, y_l) and the correspondence p_r with coordinates (x_r, y_r) as $d_x = x_l - x_r$.

A correlation matching method is used to calculate the stereo correspondences between pairs of rectified images. To this end, a square area centered on the point of interest (x_l, y_l) , called the template, is selected from the *current rectified image*₁. The template size is equivalent to the area occupied by a person in the captured image.

Similarly, in the current right-hand image *current rectified edge image*₂, an area is selected in which the correspondence may be located. This area is defined by the point of interest using two constants, d_1 and d_2 , which limit the disparity values possible to within the range $\{x_l - d_2, x_l - d_1\}$, as can be seen in Fig. 2. These limits are related to the position of the system and the floor, respectively.

The sum of absolute differences [see (3)] is used to obtain the matching values V_{SAD} . This method requires less t_{exe} than other methods while obtaining optimum results when calculating the disparity map. Correspondence is established at the minimum value $V_{\text{SAD min}}$ of those calculated for an area; consequently, $d_x = x_l - x_{V_{\text{SAD min}}}$.

$$V_{\text{SAD}} = \sum_{j=0}^{m-1} \sum_{i=0}^{n-1} \text{abs} [I_1(i, j) - I_2(i, j)] \quad (3)$$

where m is the number of rows and n is the number of columns of the region of interest.

The size of the template and the features, such as edges, used in the matching process provide a moderate range of values in which to place the correspondence. In other words, it is possible to evaluate the difference between people and flat objects.

C. Extended Particle Filter With Random Reinitialization

Each individual tracking can be considered as a single target with its own nonlinear discrete time system. Different targets may be updated in the same sampling time, and therefore, we propose the use of a multimodal estimator. The extended particle filter proposed for our system is based on the *extended condensation algorithm* to present the multimodal estimator, proposed by Kolle-Meier and Ade [26].

This particle filter is a probabilistic recursive estimator, the operation of which is based on the discrete representation of the posteriori probability density function (*pdf*), expressed as $p(\vec{x}_t | \vec{z}_{1:t})$, through a set of particles. This set of particles is represented as $S_t = \{\vec{s}_{i,t} = \vec{x}_t^{(i)}\}_{i=1}^N$.

Each particle has an associated normalized weight $\tilde{w}_t^{(i)}$, such that $\sum_{i=1}^N \tilde{w}_t^{(i)} = 1$, to characterize its probability within a global *pdf*. Thus, the output particle filter is represented as

$$p(\vec{x}_t^{(1:N)} | \vec{z}_{1:t}) \cong S_t \cong \{\vec{x}_t^{(i)}, \tilde{w}_t^{(i)}\}. \quad (4)$$

This approach is achieved by applying *Monte Carlo* sampling, where the posteriori *pdf* can be approximated as

$$p(x_t | z_{1:t}) \approx \sum_{i=1}^N \tilde{w}_t^{(i)} \delta(x_t - x_t^{(i)}). \quad (5)$$

If the approximately distributed weighted set was generated according to a *a priori pdf* $p(x_{t-1} | z_{1:t-1})$ and the new distribution is obtained from a $p(x_t | x_{t-1}, z_t)$, the weight calculation can be carried out as in (6). When the number of particles increases, the approximation to $p(x_t | z_{1:t})$ is better represented so that a certain number of particles is required for proper operation.

$$\tilde{w}_t^{(i)} \propto \tilde{w}_{t-1}^{(i)} \frac{p(z_t | x_t^{(i)}) p(x_t^{(i)} | x_{t-1}^{(i)})}{p(x_t^{(i)} | x_{t-1}^{(i)}, z_t)}. \quad (6)$$

Degeneration problems occur in the particle filter, and thus, to reduce this effect, a resampling step is included [27]. Finally, a reinitialization stage is incorporated in order to perform multihypothesis tracking and to be able to add new hypotheses to the *a priori pdf*. The stages of the extended particle filter with random reinitialization are explained hereinafter.

1) *Initialization*: Initially, the entire set of particles S_t is distributed throughout the counting area using independent identical distribution (i.i.d.). All the particle weights $\tilde{w}^{(i)}$ are initialized at the same value.

2) *Propagation*: Each particle in the set is propagated at the subsequent time instant using the state or updating model, as cited in [28]. The general expression is given in

$$p(\vec{x}_t | \vec{z}_{1:t-1}) = \int p(\vec{x}_t | \vec{x}_{t-1}) \cdot p(\vec{x}_{t-1} | \vec{z}_{1:t-1}) d\vec{x}. \quad (7)$$

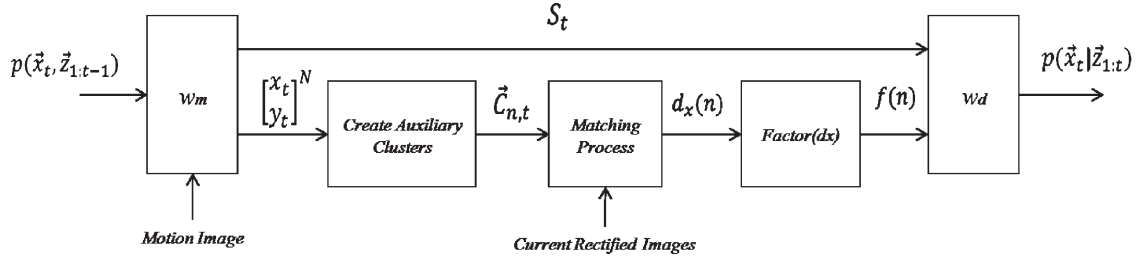


Fig. 3. Observation stage in the extended particle filter with random reinitialization.

Specifically, the motion tracking model presented in the algorithm given is a model of constant velocity which corresponds to

$$\begin{aligned} \vec{x}_t &= A\vec{x}_{t-1} + \vec{w}_{t-1} \\ \begin{bmatrix} x_t \\ y_t \\ v_{x,t} \\ v_{y,t} \end{bmatrix} &= \begin{bmatrix} 1 & 0 & t_s & 0 \\ 0 & 1 & 0 & t_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ v_{x,t-1} \\ v_{y,t-1} \end{bmatrix} + \begin{bmatrix} w_{x,t-1} \\ w_{y,t-1} \\ w_{v_{x,t-1}} \\ w_{v_{y,t-1}} \end{bmatrix} \\ \vec{y} &= C\vec{x}_t + \vec{o}_t \end{aligned} \quad (8)$$

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_t \\ y_t \\ v_{x,t} \\ v_{y,t} \end{bmatrix} + \begin{bmatrix} o_{x_t} \\ o_{y_t} \end{bmatrix} \quad (9)$$

where $\vec{x}_t = [x_t \ y_t \ v_{x,t} \ v_{y,t}]'$ is the proposed state vector used in the algorithm. This state vector consists of x_t, y_t , which represents position, and $v_{x,t}, v_{y,t}$, which represents particle velocity. The parameter t_s indicates the time between consecutive samples, and w and o are random uncorrelated noise vectors which characterize possible velocity or position variations, respectively. The velocity components of each particle should be such that they characterize the velocity vector of the hypothesis reflected in the *a priori* belief. For each iteration, these velocity components are updated with the velocity given in the hypothesis corresponding to the particle. The weight associated with each particle is maintained equal to its weight prior to predicting its state vector.

3) *Observation*: Fig. 3 shows a block diagram of the observation stage. Equation (10) expresses the weight function $w^{(i)}$, which represents the observation model applied to each particle

$$w^{(i)} = \alpha \cdot w_m^{(i)} + \beta \cdot w_d^{(i)}. \quad (10)$$

This function is a weighting between two factors, shown in Fig. 4:

- 1) amount of motion in the *ROI* of the particle ($w_m^{(i)}$);
- 2) estimated depth of the 3-D voxel belonging to the particle with respect to the camera ($w_d^{(i)}$).

Each of these factors has a fixed percentage of influence on the weight function, determined by α and β [see (10)]. Motion in the image is a necessary condition, indicating that an object or person is moving through the counting area; thus, the value of α should be greater than that of β . If α is too high and, consequently, if β is too low, the particles will congregate in areas presenting a large amount of movement, without affecting the height hypothesis in the equation. This leads to false detections such as shopping trolleys, bags, etc. Therefore, the

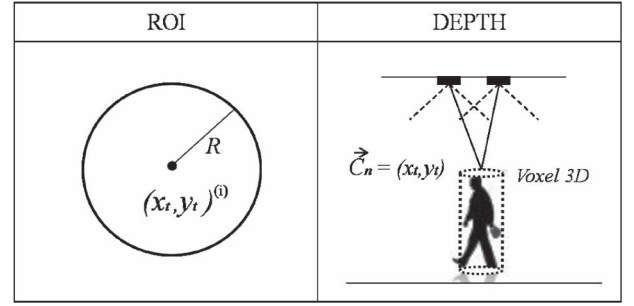


Fig. 4. Factors affecting the weight of each particle: Region-based optical flow and depth.

best performance ratio is where the α and β values are similar, maintaining $\alpha > \beta$. Various tests using different pairs of values were carried out, establishing the values for α and β as $\alpha = 0.6$ and $\beta = 0.4$.

First, for each particle, the partial weight referring to the amount of movement ($w_m^{(i)}$) is calculated as the normalized sum of all pixels of image motion located within a circular region of interest

$$w_m^{(i)} = \frac{\sum I_m(x, y)_{|ROI}}{[\sum I_m(x, y)_{|ROI}]_{\max}}. \quad (11)$$

This *ROI* is centered on the coordinates (x_t, y_t) of the particle $\vec{x}_t^{(i)}$ in question, with radius R . The radius R is a fixed parameter in the system which depends on the area that a person occupies in the image. Variation in this area depends on the relationship between the image pixel size, type of lens, and height of the stereo system above floor level. This type of *ROI* enabled us to concentrate the particles in the center of the area containing movement.

Moreover, to calculate $w_d^{(i)}$, it is necessary to obtain the depth of each particle with respect to the stereo system. However, calculating all the correspondences (number of particles in the tests $N = \{200, \dots, 2000\}$) would have implied an extremely high computational cost, thus drastically reducing the execution frame rate and eliminating the possibility of real-time execution. In order to solve this problem, we opted to select only those particles which exceeded the motion threshold THR_m , generating a subset of particles S'_t . All particles belonging to the subset S'_t thus generated are grouped into auxiliary clusters $\vec{C}_{1:n,t} = [x_t \ y_t]'$ by a distance condition. This clustering process will be described hereinafter. Each auxiliary cluster $\vec{C}_{n,t}$ represents a cylindrical 3-D voxel identified by the mass center

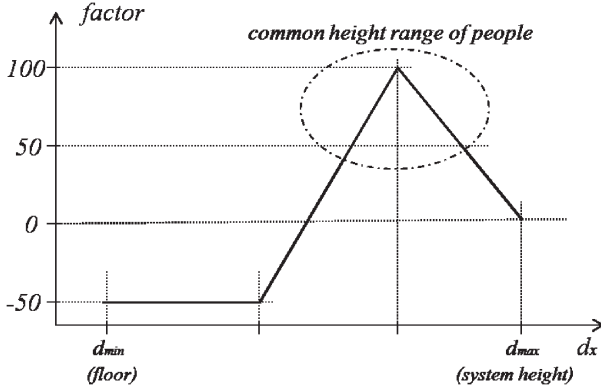


Fig. 5. Profile function of disparity.

$\vec{x}_t^{(i) \prime}$ of its component particles. To calculate the mass center, (12) is used

$$\vec{C}_{n,t} = \frac{1}{M} \sum_{i'} w_m^{(i) \prime} \vec{x}_t \quad (12)$$

where $M = \sum_{i'} w_m^{(i) \prime}$.

Disparity $d_x^{(n)}$ for each cluster $\vec{C}_{n,t}$ with respect to the stereo system is determined. In this way, each particle in the subset S_t' is assigned the corresponding weight $w_d^{(i)}$ in accordance with the disparity value $d_x^{(n)}$ of the cluster $\vec{C}_{n,t}$ to which it belongs and the distance $d_C^{(n,i)}$. The distance $d_C^{(n,i)}$ represents the *Euclidean* distance between the position of the particle $\vec{x}_t^{(i) \prime}$ and the position of the corresponding mass center $\vec{C}_{n,t}$. Thus, $w_d^{(i)} = f(d_x^{(n)}, d_C^{(n,i)})$, as explained by

$$w_d^{(i)} = \underbrace{\left(1 + \frac{\theta - 1}{d_{\max}} d_C^{(m,i)}\right)}_{wd_1} \times \underbrace{\text{factor}\left(d_x^{(m)}\right)}_{wd_2} \quad (13)$$

where θ is a constant which can be defined between $\{0,1\}$ and d_{\max} is the maximum distance that a particle can have from the mass center in order to form part of a cluster. wd_1 reduces the weight of a particle the further it is from the mass center of the cluster to which it belongs, while θ establishes the amount by which to reduce the weight. wd_2 provides a factor according to the cluster's disparity value $\vec{C}_{n,t}$. These factors are tabulated since the calculation is carried out offline. Fig. 5 shows the proposed profile applied. A factor influencing the particle weight is assigned to each disparity value. For disparity values within the most common height range of people (1.6 m, 2.0 m), the factor is large. However, for disparity values corresponding to heights below 1.0 m (objects), the factor decreases. An area of uncertainty is included between the two ranges. Finally, possible disparity values outside the range $\{d_x(0 \text{ m}), d_x(2.1 \text{ m})\}$ are assigned the value $f(d_x) = -50$.

The particles $\vec{x}_t^{(i)}$ whose values $w_m^{(i)}$ do not exceed the threshold THR_m , i.e., which do not belong to the particle subset S_t' , are assigned a $w_d^{(i)} = 0$. Furthermore, where there are not enough features in the template or the value of correspondence is very low, we propose $w_d^{(i)} = 0$ for the particles belonging to a cluster.

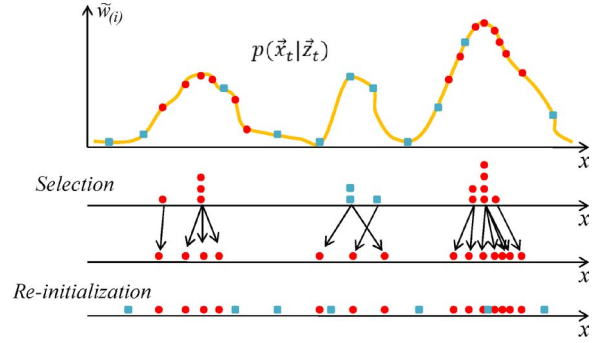


Fig. 6. Working example of the stages of selection and reinitialization.

The fact of not applying a factor to particles which present little movement does not give rise to detection or tracking problems since the global weight is already low for the same reason. As indicated earlier, the existence of movement in the area is an *a priori* condition since areas without movement are irrelevant to the system.

4) *Selection*: Conducting a resampling process minimizes the problems of degeneration of the particle set. We used multinomial resampling, as proposed in [29]. A fixed number of particles $N - M$ is selected in order to carry out the selection stage, where N is the total number of particles and M is the number of new particles incorporated in the reinitialization stage. For the selection process, a uniform random multinomial resampling of the discrete belief $p(\vec{x}_t | \vec{z}_{1:t})$ is carried out, which provides the correction stage, as can be seen in Fig. 6. It is necessary to normalize all particle weights $w^{(i)}$ using the expression in (14). In this way, more of the particles with greater weight $\tilde{w}^{(i)}$ are regenerated than particles presenting a lower weight $\tilde{w}^{(i)}$, as can be seen in Fig. 6.

$$\tilde{w}^{(i)} = \frac{w^{(i)}}{\sum_{i=0}^N w^{(i)}}. \quad (14)$$

This selection process involves another problem in filter operation, known as the impoverishment of the particle set. Particles with a high weight are selected many times, generating new particles around the same position. To reduce this effect, we propose introducing a random element into the position components $[x_t, y_t]$ when a particle is regenerated. This random element is included in the state vector x_t in order to obtain more filter diversity since different particles located at the same position do not improve the filter's efficiency but rather provide redundant information, thus implying that a number of particles in the set are wasted. However, the particle retains the same weight $\tilde{w}^{(i)}$.

5) *Reinitialization*: In the *extended condensation algorithm*, the M particles included in this stage are strategically added to possible new hypotheses based on the measurement vector $\vec{z}_{1:t-1}$ obtained in the time $t - 1$. This is made possible by obtaining the measurement vector without any *a priori* information about the particle set. If this were not the case, i.e., if the measurement vector has been obtained using information about the set, it would not be possible to obtain new hypotheses without some particles reverting to their former position in the said hypotheses.

Therefore, to complete the set S_t in the proposed particle filter, M particles are randomly initialized using i.i.d., without any kind of *a priori* information and with the weight $\tilde{w}^{(i)} = 0$. The M number of particles is maintained constant throughout the execution of the filter, and using this section of the set, new hypotheses are incorporated into the $p(\vec{x}_{t-1}|\vec{z}_{1:t-1})$. In order not to discard any new hypothesis, it is necessary for the value of M to be sufficiently high with respect to the total number of particles comprising the set. In this way, we ensured that a sufficient number of particles are distributed throughout the counting area. This may imply that, when there is a hypothesis, the particle filter's efficiency value at each iteration might be insufficient, thus producing the degeneration of the set. In [30], the parameter \tilde{n}_{eff} is established to measure the particle filter efficiency, in which it is necessary to obtain the value $\tilde{n}_{\text{eff}} \geq (2/3)N$. To calculate \tilde{n}_{eff} , the expression shown in (15) is used

$$\tilde{n}_{\text{eff}} = \frac{1}{\sum_{i=1}^N \left(\tilde{w}_t^{(i)}\right)^2}. \quad (15)$$

In order to ensure compliance with the previous relation, M is established as $M = (1/3)N$ so that the remaining $N - M = (2/3)N$ are used for the selection stage. In this way, $2/3$ of the set at each iteration represents the $p(\vec{x}_{t-1}|\vec{z}_{1:t-1})$, ensuring particle filter efficiency and thus eliminating the possibility of degeneration of the set. Fig. 6 shows an example of the *selection* and *reinitialization* stages. First, in the selection stage, the resampling process is carried out. As can be observed, particles which have a high weight are randomly regenerated. Subsequently, in the reinitialization stage, different particles (blue squares) are added in order to introduce new hypotheses.

D. Clustering Method

Using the probabilistic solution generated by the particle filter, it is necessary to include some method of association in order to express the result as a deterministic output. Once the $p(\vec{x}_t|\vec{z}_{1:t})$ has been obtained, all the set particles which exceed the threshold weight THR_w are grouped to form different cylindrical 3-D voxels using particle clusters $\vec{C}_{n,t} = [x_t \ y_t]'$, which represent each belief hypothesis. The parameter THR_w is set at 0.5 within the normalized range $\{0,1\}$. This grouping is carried out prior to the selection stage since the vectors \vec{x}_t of each particle have not been modified and represent the discrete belief in time t .

This process is based on a standard K-means algorithm [31], which groups particles into clusters, minimizing the quadratic distance between each position of a particle and the cluster centroid. However, it needs to know *a priori* the number of clusters to generate, an awkward requirement to meet in our case since the number of 3-D voxels is not known *a priori*. Therefore, we incorporated a modification into the algorithm, shown in [20]. With this modification, we eliminated the need to give an *a priori* number of clusters, but it is necessary to incorporate a parameter of maximum distance d_{max} between clusters in order to distinguish between different voxels. This parameter represents the maximum distance at which a cluster particle can be located in order to form part of the correspond-

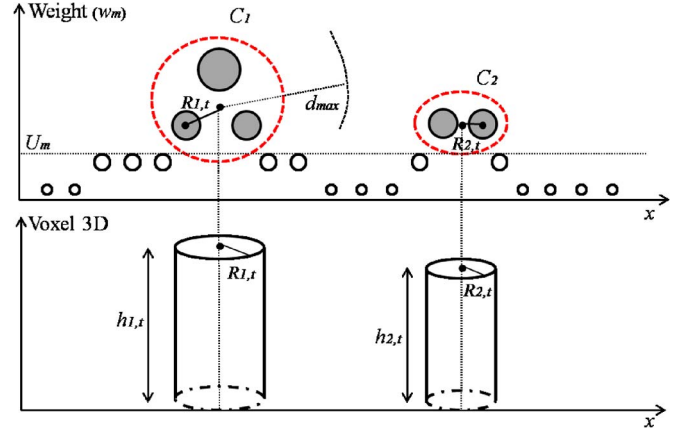


Fig. 7. Example of clustering method and 3-D voxel representation.

ing cluster. Fig. 7(a) gives an example, where the belief is represented in two clusters since not all the particles fulfilled the maximum distance condition, providing a deterministic output of two cylindrical 3-D voxels ($\vec{V}_{n,t}$). The radius $R_{n,t}$ of the 3-D voxels is the distance between the particle furthest away from the 3-D voxel centroid, and the height $h_{n,t}$ is the system height minus depth. The depth is calculated using the mean weight of the factors (w_{d2}) of each particle belonging to the cluster in question.

In order to generate clusters, we selected the coordinates of a random particle $\vec{x}_t^{(i)} = [x_t \ y_t]'$ to form the center of the first cluster $\vec{C}_{1,t}$, in the case where there were no clusters from the previous iteration. However, where they do exist, all clusters in $(t-1)$ are used to regenerate clusters in (t) . All particles are associated with their closest cluster according to the maximum distance condition, expressed in

$$\begin{cases} d_{\min_{\vec{C}_{n,t}|\vec{x}_t^{(i)}}} \leq d_{\max} & \vec{x}_t^{(i)} \rightarrow \vec{C}_{n,t} \\ d_{\min_{\vec{C}_{n,t}|\vec{x}_t^{(i)}}} > d_{\max} & \vec{x}_t^{(i)} \rightarrow \text{new } \vec{C}_{n,t} \end{cases} \quad (16)$$

where $d_{\min_{\vec{C}_{n,t}|\vec{x}_t^{(i)}}}$ is the minimum distance between the position of the particle and the existing clusters.

When a particle does not fulfill the previous condition, it is assigned to the centroid of another new cluster. Each time a new cluster is generated, a new association between all the particles and the existing clusters is initiated. When all the particles have been assigned to the different clusters, centers, radius, and heights are determined, generating a set of detections (3-D voxels) which represent the hypotheses detected in the time t [see Fig. 7(b)] and, in turn, are represented by the variables given in (17). Any cluster without an associated particle is eliminated.

$$\vec{V}_{n,t} = \begin{bmatrix} \vec{C}_{n,t} = [x_t \ y_t]' & \text{voxel center} \\ R_{n|t} & \text{voxel radius} \\ h_{n|t} & \text{voxel height} \end{bmatrix} \quad (17)$$

This algorithm is also used to provide the auxiliary clusters shown in the *observation* stage.

E. Tracking and Counting People

To perform a count, it is necessary to construct trajectories using the detections ($\vec{V}_{n,t}$) generated at each particle filter iteration. These trajectories are generated using a data association method known as the *nearest neighbor (NN)*, a method commonly used for simple association problems in which the data present the minimum of interactions, as in the present case, and where the detection of different trajectories does not fluctuate over consecutive iterations. The overhead camera location leads to a reduction of occlusions, a problem associated with systems using tilted cameras. Furthermore, it yields very low execution times, which is of great interest in the proposed system. Each tracking $\vec{T}_{m,t}$ stores the detections $\vec{V}_{n,t}$ assigned by the *NN* method and a velocity vector $\vec{v}_{m,t} = [v_x \ v_y]'$, which is updated at each iteration to add the new detection. The calculation of the distances between trajectories and detections, in the *NN*, is carried out using the Euclidean distance of the characteristics that form a detection, as shown in

$$d_{\vec{T}_{m,t-1}, \vec{V}_{n,t}} = \sqrt{(\Delta x)^2 + (\Delta y)^2 + (\Delta h)^2}. \quad (18)$$

As can be observed in (18), the parameter R is not used since it no longer provides any useful information and is only used to depict the results of the tests.

Any detections, from any iteration, which are not associated with existing trajectories initiate a new tracking. To provide tracking generation with robustness in this stage, a tracking lifetime algorithm is added, together with another algorithm for consecutive detection losses in a tracking. The lifetime of trajectories to which a detection has been added is increased by one unit up to a fixed limit or is similarly decreased if no detection is added. If a trajectory is detected as not having included a detection for a consecutive number of times or if its lifetime has reached zero, it is considered finished and is eliminated. Eliminated trajectories are no longer included in the people count. Inputs and outputs are counted according to the start and the finish point of a tracking. The crosslines include various articles [32] in order to carry out counting and distinguish between an entrance and an exit. Part of the error in these systems arises from trajectories which do not cross one of these lines and thus are not added to the count. For this reason, in the proposed system, we chose to use the comparison of the distance (in length) traveled by a tracking with the threshold length. This distance is determined by the difference between the last detection added and the first. Thus, if the absolute value of the tracking distance is greater than or equal to the threshold length, the tracking is considered valid. The sign of the difference indicates whether the tracking corresponds to an entrance or an exit.

As mentioned in the *propagation* stage, the velocity components $[v_{x,t} \ v_{y,t}]$ of each particle are updated at each iteration in order to characterize the velocity of the hypothesis that they represent. For each tracking, the velocity $\vec{v}_{n,t+1}$ is determined for the subsequent propagation stage using the current detections $\vec{V}_{n,t}$ and $\vec{V}_{n,t-1}$ and velocity $\vec{v}_{n,t}$. The latter is included in order to smooth changes in direction and module. In the case of new trajectories with only one associated detection, the velocity of

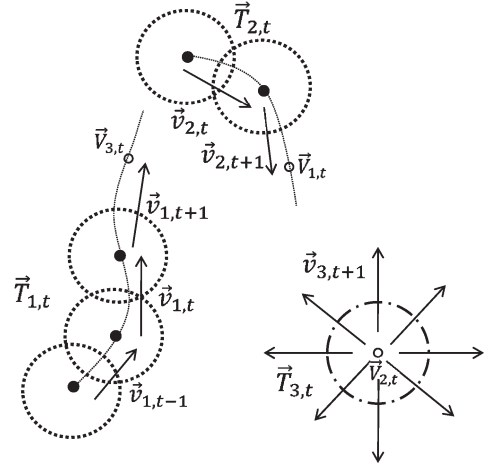


Fig. 8. Proposed hypothesis tracking.

the particles associated with this detection is randomized using i.i.d. in all directions. Fig. 8 gives an example where two detections ($\vec{V}_{1,t}$ and $\vec{V}_{3,t}$) are associated with existing trajectories ($\vec{T}_{1,t}$ and $\vec{T}_{2,t}$) using *NN*, and a detection $\vec{V}_{2,t}$ generates a new tracking $\vec{T}_{3,t}$ as it has not been associated with any existing tracking.

IV. RESULTS

A. Test Platform

In this paper, we have presented a solution composed of two static low-cost cameras. Positioning is adaptable within different urban environments, such as buildings where lighting is maintained at an almost constant level while the system is running. In the majority of cases, the system can be installed directly onto the ceiling of passageways. Where this is not possible, it can be affixed to a post to provide the height required by the system, as shown in Fig. 9(a). The height will be imposed by site restrictions. The proposed system can operate at different heights, after making the necessary adjustments to the system parameters, without a reduction in its hit rate. There are two requisites for satisfactory people detection: the presence of optical flow and correct correspondence. People should cross the counting area at a speed greater than that imposed by the motion threshold (THR_m). An exact correspondence is not necessary to carry out robust tracking of the person because a disparity value is later applied to a weight function that determines the overall value of the particles of the same voxel. In addition, the maximum distance parameter (d_{\max}) used in the clustering method determines the number of possible voxels. This parameter is adjusted depending on the space occupied by a person in the image. An incorrect setting of this parameter could increase false positives if the parameter is too small or increase false negatives if the parameter is too large, which influences the sensitivity of the final results. The algorithm parameters, such as the maximum number of people to count at a time, motion threshold, lifetime, and maximum distance between clusters, are determined from system parameters, such as frames per second (fps), image size, area covered, and system

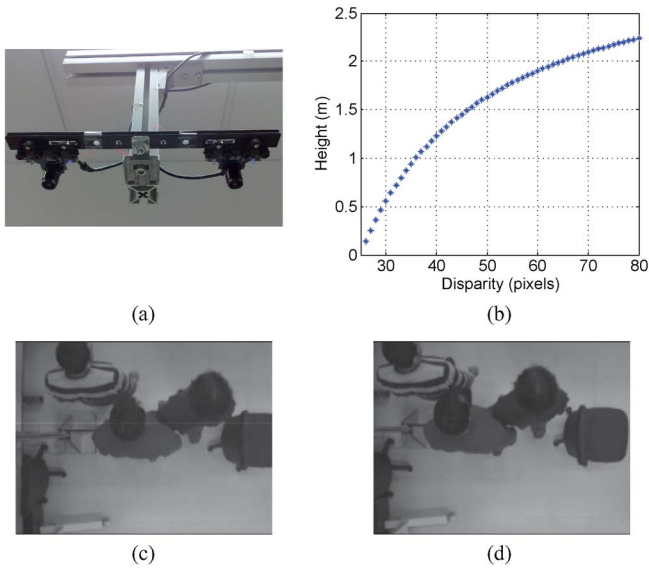


Fig. 9. Test platform. (a) View of prototype to validate proposed system. (b) Height values for the possible values of disparity. (c) Rectified left image. (d) Rectified right image.

height, and also require a minimum processing capacity of the computer platform.

The distance between cameras is another parameter that affects the error rate if it is not adjusted properly. To achieve a minimum error rate, it is necessary to impose a distance so that the area with the highest resolution is set to a common height range for people. Fig. 9(b) shows different height values depending on the possible values of disparity. It can be seen that the range of disparity values with the highest resolution is set to common height values.

Specifically, our cameras are located in an overhead position, about 3 m above floor level, fixed to a mount located on an aluminum post. The lens of the sensor used has a focal length of about 5 mm. With these characteristics, an effective surface area of about 3 m² is achieved for tracking. Thus, a maximum of 4/5 people can be counted simultaneously. In the test video acquired, both sensors were configured to a speed of 30 fps and a resolution of 320 × 240 pixels. The algorithm was codified in C++ language using open source libraries (Open CV 2.1) and was executed without multiple-thread support. All experiments were performed with an embedded MINI-ITX PC (Atom 1.66 GHz). Fig. 9(c) and (d) shows a pair of rectified stereo images.

B. Analysis of Proposed Particle Filter

Different practical tests were performed to verify the functionality of the proposed particle filter. Important aspects for this type of algorithm based on particle filters, such as execution time, frame rate, efficiency, and deterioration problems, are analyzed thoroughly in this section.

1) *Execution Time*: Fig. 10 presents the execution time expressed in fps and the error rate expressed as a percentage of the overall count, depending on the number of particles using the extended particle filter proposed. Two experiments are shown,

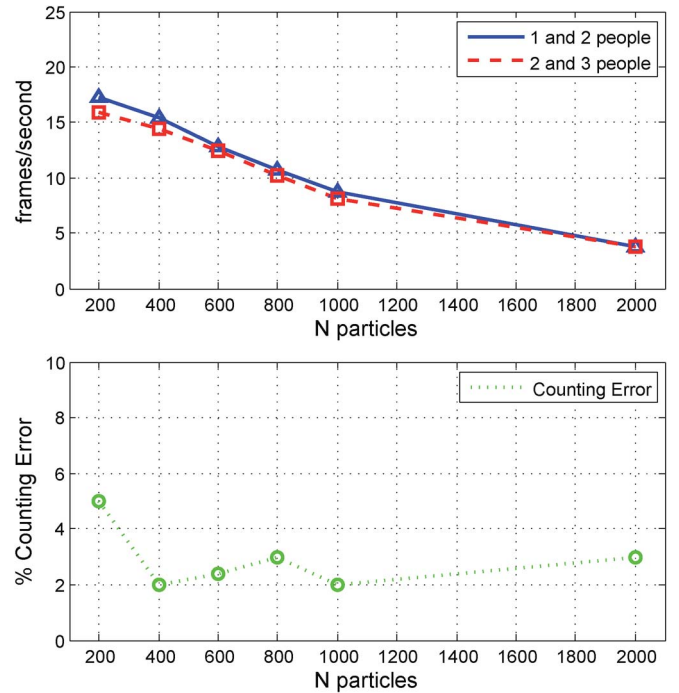


Fig. 10. Execution time and error rate according to the number of particles in the proposed extended particle filter.

each of which was performed using 1000 images with motion information, considering interactions between 1/2 people and 2/3 people in each case. As can be seen, the execution time was almost independent of the number of people crossing the counting zone. Small variations were due to the increase in matching processes performed in the algorithm.

The variation in error rate was negligible in the experiments, showing that, even under complex testing situations, an increase in N does not yield any improvement in the reliability of the estimate. Nevertheless, a minimum number of particles is necessary for this compound set in order for the application to function correctly.

In contrast, a variation in N implied a significant change in the run time. In general terms, it follows that incorporating an increase in the $\Delta N = 100$ set of particles leads to a decrease of 1.25 fps. Given the operational conditions and the frame rate, it is necessary to use one value or another of N .

2) *Efficiency*: Here, we present the instantaneous values of \hat{n}_{eff} generated by the proposed filter. A video test with continuous crossings of two people was used. Fig. 11 includes the results for different values of particles (N), confirming that an increase in N does not imply an improvement in the efficiency value. Basically, \hat{n}_{eff} increases by approximately 1% when N is increased by 50%.

Different circled areas can be seen where instantaneous values presented very low efficiency. This happens when a person enters or leaves the count area. The first motion detections correspond to the lower extremities of the person. These areas present a height that identifies them as an object so that the weight $w_d^{(i)}$ to acquire the particles that make up the cluster for evaluation is very low, even negative. Consequently, instantaneous values of very low efficiency are obtained in those

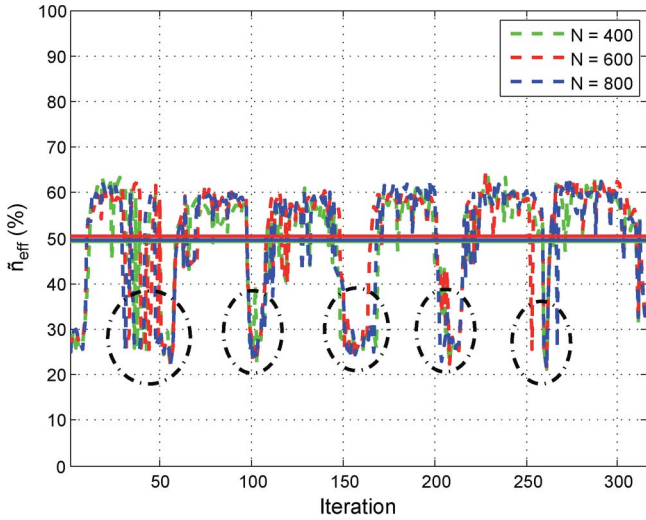


Fig. 11. Comparison of the efficiency for three values of N (400, 600, and 800). A line is added to represent the average efficiency over several iterations.

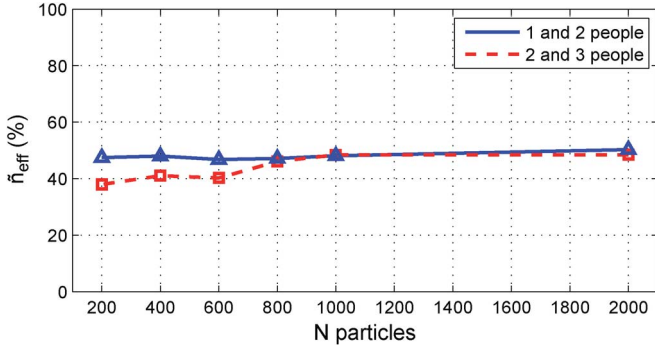


Fig. 12. Average standard efficiency based on the number of particles used in the proposed filter.

iterations where a person enters or leaves the count zone. In the event that other people cross the count area at the same time, the instantaneous value of efficiency does not decrease, given that fewer particles are concentrated in that area to represent the person. Thus, it has been demonstrated that the selection stage presents a correct operation.

In addition, there are variations in the instantaneous values of efficiency due to problems in the search for correspondence. Several situations may arise where there are not enough features in the regions used to perform the matching, so the factor applicable to the cluster has a low value, affecting all particles comprising the cluster. Therefore, the instantaneous value \tilde{n}_{eff} in that iteration decreases. Thus, these cases should not be understood as the degeneration of the set of particles.

Fig. 12 presents a graph showing the average value of instantaneous efficiency values \tilde{n}_{eff} according to the number of particles N , under the same test conditions as those described earlier. It can be seen that increasing the number of particles does not provide a significant improvement in efficiency values.

3) *Deterioration Problems*: Fig. 13 presents a percentage value of \tilde{n}_{eff} and the number of hypotheses added to the belief according to the number of particles used for the selection stage. The results were obtained from a video test with crossings of

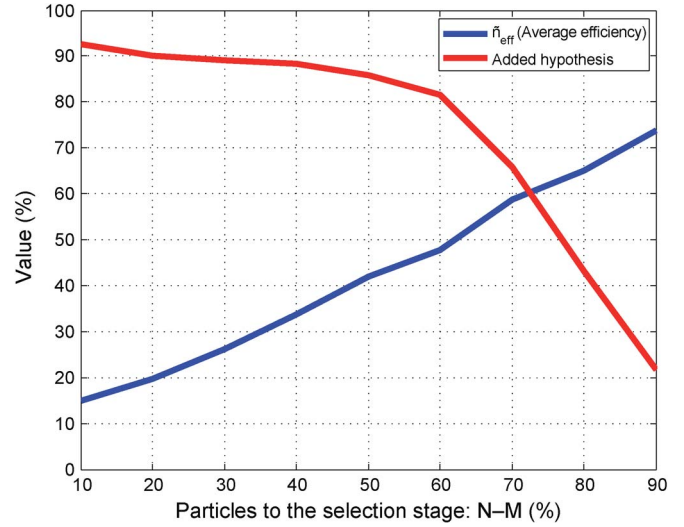


Fig. 13. Average efficiency value based on the percentage of particles for the current selection stage.

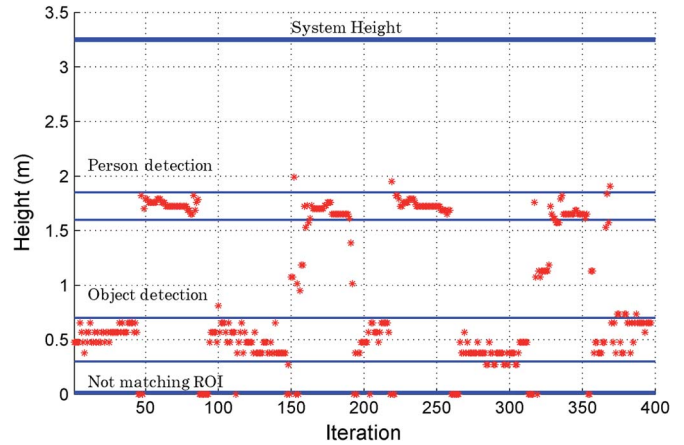


Fig. 14. Height measurement test consisting of an object and a person.

three people and a value of N particles equal to 400. As can be seen, \tilde{n}_{eff} increases with the percentage of particles used in the selection stage ($N - M$) since a higher number of particles is used to represent the belief or *posteriori pdf*. In contrast, the number of hypotheses that should be added to the belief decreases due to the small percentage of particles used to add new hypotheses at random. In this situation, high error rates are obtained in the counting results. For proper system operation, the value of M selected should ensure higher efficiency values and include all possible people candidates.

4) *Test of Stereoscopic Measures*: Fig. 14 presents a test to measure the height with the proposed stereoscopic configuration proposal. The test consisted of a set of 400 images in which a person and an object counting were continuously performed through the area. It is shown that the accuracy decreases with increasing depth due to the inherent properties of the configuration. Different situations are presented in Fig. 14.

- 1) Measures that represent the passage of a person counting through the counting area: They are located within the area *Person detection*.

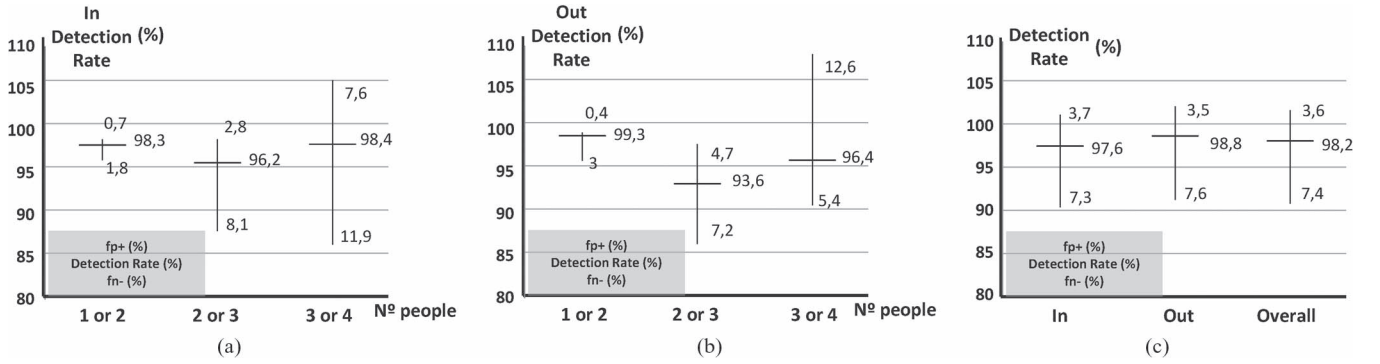






Fig. 15. Results of the proposed system in different situations. (a) In. (b) Out. (c) Overall.

TABLE I
COUNT RATE IN DIFFERENT SYSTEMS

Teixeira [33]	Snidaro [14]	Proposed System	Albiol [34]
79,5%	96,25%	98,2%	98,7%
			

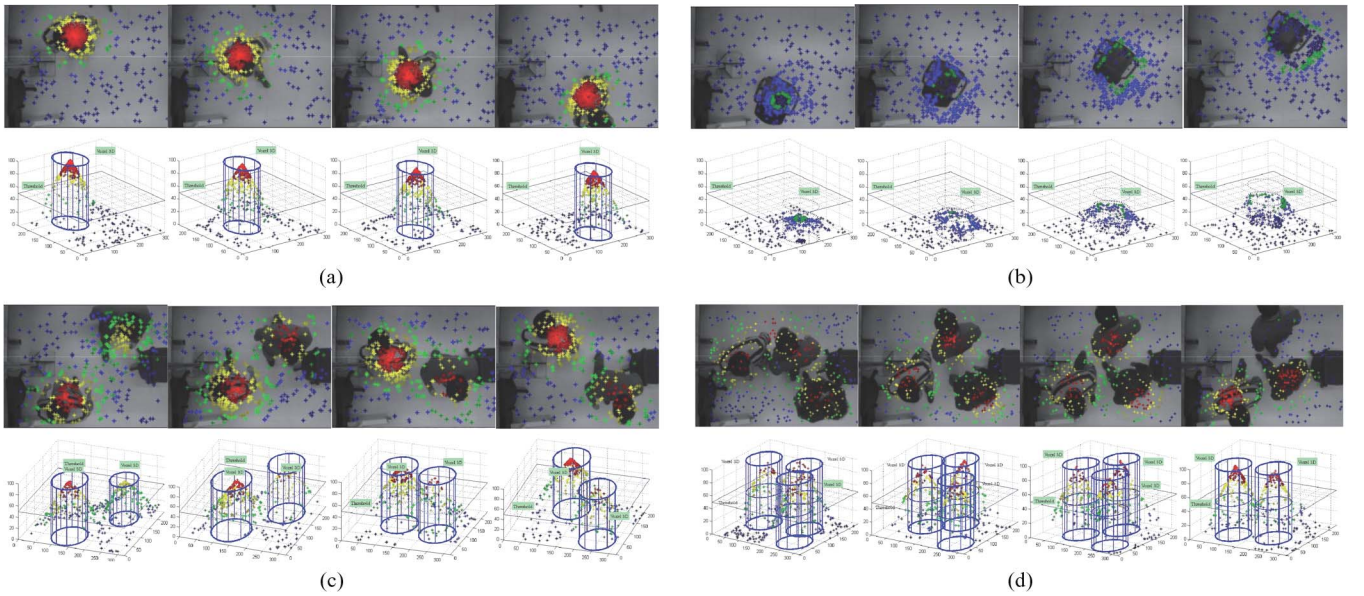


Fig. 16. Some experimental results of people counting. (a) Person. (b) Object. (c) and (d) Multiperson.

- Measures that represent the passage of an object by counting through the counting area: They are located within the area *Object detection*.
- Measures that represent no matching: They are at the bottom of the figure.

C. Counting Results

The system was positioned in a passageway where different situations arose, such as people moving in different directions

or groups of up to four people moving through the counting area. Fig. 15 gives the results obtained from the counting system in different scenarios. The results have been extracted from a total of 300 people, divided into three types of crossings: 1/2 people, 2/3 people, and 3/4 people. Each type of crossing is represented by three values: detection rate, false positives (fp^+), and false negatives (fn^-), all expressed in %. The detection rate represents the counting system, fn^- represents the people that were not counted, and fp^+ represents the false counting of inexistent people, all with respect to actual people in each category.

As can be seen, the stereo counting system presented a certain percentage of error. These errors are due to different reasons, as explained in the following.

- 1) Detection problems: The lack of contrast between the floor and the person moving through the counting area produced a low difference between successive images which was not detected, increasing the rate of false negatives.
- 2) Slow movement problems: People stopped in the counting area, or presented very slow movement, generating insufficient optical flow. Where intervals of consecutive images occurred (stop-and-go), which contain detectable movement, a valid count was always obtained. Otherwise, the individual was not counted.
- 3) Deterioration problems: These may occur when multiple people (more than four) interact through the counting area. The subset $N - M$ in the selection stage requires a larger number of particles to represent the posteriori *pdf*, as low efficiency values are obtained. Thus, in the clustering method, some hypotheses are not identified. Note that the detection rate decreased when there was an increase in the ratio number of people/area.

Table I shows the overall average results for the count rate provided by different counting systems proposed in the literature. In [33], a sensor network is implemented, composed of multiple nodes with partial overlapping in order to obtain a larger counting area. Each node is configured at 320×240 pixels and is located 2.4 m above the floor. Similar parameters are used in [14]. This counting system consists of a single sensor located 3 m above the floor which acquires images with a resolution of 384×288 pixels. In the same way, Albiol *et al.* [34] propose a system aimed at identifying the number of people in a train carriage. Images are acquired with a resolution of 756×576 pixels from the top of the door. An example image from each system is shown in Table I.

To conclude the results section, Fig. 16 presents some examples of system operation in different situations. In each subfigure, the actual situation of each particle overlaps with a color defined according to weight. At the bottom of each subfigure, there is a 3-D representation of the set of particles and voxels representing the hypothesis.

V. CONCLUSION

In this paper, we have presented a new proposal for bidirectional counting based on images from a stereoscopic overhead view camera system.

The extended particle filter with random reinitialization provides the probabilistic and multimode characteristics required to carry out multiple-hypothesis tracking. The modified K-means clustering method is incorporated in order to provide deterministic output. Stereovision is a key element for differentiating between people and other objects that may appear in the count. A minimum degree of movement is required for human motion to be considered detectable motion. Several types of movements, such as stop-and-go, are processed satisfactorily.

The main contribution of this paper is the inclusion of different features relevant to people tracking (movement, size,

and height), adapting a particle filter followed by the implementation of a clustering method, providing robustness to the algorithm. The reinitialization stage of the proposal is capable of incorporating new hypothesis beliefs. This stage provides a constant execution time regardless of the number of hypotheses. The proposed algorithm presents problems of particle set deterioration when many people (more than four people) interact, crossing the counting area at the same time.

REFERENCES

- [1] S. Velipasalar, Y.-L. Tian, and A. Hampapur, "Automatic counting of interacting people by using a single uncalibrated camera," in *Proc. IEEE Int. Multimedia Expo Conf.*, 2006, pp. 1265–1268.
- [2] A. Chan and N. Vasconcelos, "Counting people with low-level features and Bayesian regression," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2160–2177, Apr. 2012.
- [3] Y.-L. Hou and G. K. H. Pang, "People counting and human detection in a challenging situation," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 41, no. 1, pp. 24–33, Jan. 2011.
- [4] X. Zhao, E. Delleandrea, and L. Chen, "A people counting system based on face detection and tracking in a video," in *Proc. IEEE Int. Conf. AVSS*, 2009, pp. 67–72.
- [5] J. D. Valle, Jr., L. E. S. Oliveira, and A. S. Britto, Jr., "People counting in low density video sequences," in *Proc. Pacific-Rim Symp. Image Video Technol.*, 2007, pp. 737–748.
- [6] L. Rizzon, N. Massari, M. Gottardi, and L. Gasparini, "A low-power people counting system based on a vision sensor working on contrast," in *Proc. IEEE ISCAS*, 2009, pp. 786–790.
- [7] H. Xu, P. Lv, and L. Meng, "A people counting system based on head-shoulder detection and tracking in surveillance video," in *Proc. ICCDA*, 2010, vol. 1, pp. V1-394–V1-398.
- [8] D. B. Yang, H. H. Gonzalez-Banos, and L. J. Guibas, "Counting people in crowds with a real-time network of simple image sensors," in *Proc. 9th IEEE Int. Comput. Vis. Conf.*, 2003, pp. 122–129.
- [9] S. Yu, X. Chen, W. Sun, and D. Xie, "A robust method for detecting and counting people," in *Proc. ICALIP*, 2008, pp. 1545–1549.
- [10] T.-H. Chen, T.-Y. Chen, and Z.-X. Chen, "An intelligent people-flow counting method for passing through a gate," in *Proc. IEEE Conf. Robot., Autom., Mechatron.*, 2006, pp. 1–6.
- [11] L. Li, S. Yan, X. Yu, Y. K. Tan, and H. Li, "Robust multiperson detection and tracking for mobile service and social robots," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 5, pp. 1398–1412, Oct. 2012.
- [12] S. X. Yan Rui and Y. Shu, "Moving object detection based on an improved Gaussian mixture background model," in *Proc. Int. Colloq. Comput., Commun., Control, Manage.*, 2009, pp. 12–15.
- [13] P. Zhang, T.-Y. Cao, and T. Zhu, "A novel hybrid motion detection algorithm based on dynamic thresholding segmentation," in *Proc. 12th IEEE ICCT*, 2010, pp. 853–856.
- [14] L. Snidaro, C. Micheloni, and C. Chiavedale, "Video security for ambient intelligence," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 35, no. 1, pp. 133–144, Jan. 2005.
- [15] F. Pernkopf, "Tracking of multiple targets using online learning for reference model adaptation," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 6, pp. 1465–1475, Dec. 2008.
- [16] C.-W. Lai, C.-M. Huang, and L.-C. Fu, "Multi-target tracking using separated importance sampling particle filters with joint image likelihood," in *Proc. IEEE Int. Conf. SMC*, 2006, vol. 6, pp. 5179–5184.
- [17] Z. Ye and Z.-Q. Liu, "Tracking human hand motion using genetic particle filter," in *Proc. IEEE Int. Conf. SMC*, 2006, vol. 6, pp. 4942–4947.
- [18] F. Ababsa and M. Mallem, "Robust circular fiducials tracking and camera pose estimation using particle filtering," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. ISIC*, 2007, pp. 1159–1164.
- [19] M. Kristan, S. Kovacic, A. Leonardis, and J. Pers, "A two-stage dynamic model for visual tracking," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 6, pp. 1505–1520, Dec. 2010.
- [20] M. Marron, M. A. Sotelo, J. C. Garcia, D. Fernandez, and D. Pizarro, "XPCFP: An extended particle filter for tracking multiple and dynamic objects in complex environments," in *Proc. IEEE ISIE*, 2005, vol. 4, pp. 1587–1592.
- [21] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

- [22] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *Proc. Int. Conf. Comput. Vis.*, 1999, pp. 666–673.
- [23] A. Donate, X. Liu, and E. G. Collins, "Efficient path-based stereo matching with subpixel accuracy," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 1, pp. 183–195, Feb. 2011.
- [24] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.
- [25] M. Asif and J. J. Soraghan, "Depth estimation and implementation on the DM6437 for panning surveillance cameras," in *Proc. 16th Int. Digital Signal Process. Conf.*, 2009, pp. 1–7.
- [26] E. Koller-Meier and F. Ade, "Tracking multiple objects using a condensation algorithm," *J. Robot., Auton. Syst.*, vol. 34, no. 2/3, pp. 93–105, Feb. 2001.
- [27] M. Bolic, P. M. Djuric, and S. Hong, "Resampling algorithms and architectures for distributed particle filters," *IEEE Trans. Signal Process.*, vol. 53, no. 7, pp. 2442–2450, Jul. 2005.
- [28] B. Liu, C. Ji, Y. Zhang, C. Hao, and K.-K. Wong, "Multi-target tracking in clutter with sequential Monte Carlo methods," *IET Radar, Sonar, Navigat.*, vol. 4, no. 5, pp. 662–672, Oct. 2010.
- [29] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *J. Amer. Statist. Assoc.*, vol. 94, no. 446, pp. 590–599, Jun. 1999.
- [30] A. Doucet, "Monte-Carlo Methods for Bayesian Estimation of Hidden-Markov Models. Application to Radiation Signal," Ph.D. dissertation, Univ. Paris-Sud, Orsay, France, 1997.
- [31] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.
- [32] J. Barandiaran, B. Murguía, and F. Boto, "Real-time people counting using multiple lines," in *Proc. 9th Int. WIAMIS*, 2008, pp. 159–162.
- [33] T. Teixeira and A. Savvides, "Lightweight people counting and localizing for easily deployable indoors WSNs," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 4, pp. 493–502, Aug. 2008.
- [34] A. Albiol, I. Mora, and V. Naranjo, "Real-time high density people counter using morphological tools," *IEEE Trans. Intell. Transp. Syst.*, vol. 2, no. 4, pp. 204–218, Dec. 2001.



Alfredo Gardel received the degree in telecommunication engineering from the Polytechnic University of Madrid, Madrid, Spain, in 1999 and the Ph.D. degree in telecommunication from the University of Alcalá, Madrid, Spain, in 2004.

Since 1997, he has been a Lecturer with the Electronics Department of the University of Alcalá. His main areas of research comprise infrared and computer vision, monocular metrology, robotics sensorial systems, and design of advanced digital systems.



Ignacio Bravo received the B.S. degree in telecommunications engineering, the M.Sc. degree in electronics engineering, and the Ph.D. degree in electronics from the University of Alcalá, Madrid, Spain, in 1997, 2000, and 2007, respectively.

Since 2002, he has been a Lecturer with the Electronics Department of the University of Alcalá. He is currently an Associate Professor with the University of Alcalá. His areas of research are reconfigurable hardware, vision architectures based in FPGAs, and electronic design.



José Luis Lázaro received the degrees in electronic engineering and telecommunication engineering from the Polytechnic University of Madrid, Madrid, Spain, in 1985 and 1992, respectively, and the Ph.D. degree in telecommunication from the University of Alcalá, Madrid, in 1998.

Since 1986, he has been a Lecturer with the Electronics Department of the University of Alcalá. He is currently a Professor with the Electronics Department of the University of Alcalá. His areas of research are robotics sensorial systems by laser, optical fibers, infrared and artificial vision, motion planning, monocular metrology, and electronics systems with advanced microprocessors.



Jorge García received the B.S. degree in telecommunications engineering and the M.Sc. degree in electronics system engineering from the University of Alcalá, Madrid, Spain, in 2009 and 2011, respectively, where he is currently working toward the Ph.D. degree in the Electronics Department.

He has been working with the Electronics Department of the University of Alcalá since 2009. His currently research interests include computer vision and system based on field programmable gate arrays.



Miguel Martínez received the degrees in telecommunication engineering from the University of Alcalá, Madrid, Spain, in 2010, where he is currently working toward the Ph.D. degree in electronic engineering in the Electronics Department.

His currently research interests include machine learning, battery modeling, battery management systems, and intervehicle communications.