

ABSTRACT

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. 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 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. It reduces communication cost and Data Retrieval is easy.

KEYWORDS: Modified condensation algorithm, Clustering,

INTRODUCTION

Crowd monitoring is very important in many aspects especially in the areas of Airports, railway stations, sports, and rallies. Excessive crowding will result in unexpected events, such as riots, fights or emergencies. Crowd density can serve as an important descriptor of crowd stability because it can quantitatively or qualitatively provide the amount of pedestrians in an area. There have been many methods for crowd density estimation using computer vision techniques. These methods can be divided into three categories,

- 1) Pixel-based analysis,
- 2) Detection-based analysis, and
- 3) Texture-based analysis.

These methods rely on specific camera training data which requires a system to be trained and tested on the same viewpoint, using potentially hundreds or thousands of annotated training frames. Even though large-scale CCTV networks are becoming increasingly common, automated crowd counting is not widely deployed.

MORPHOLOGICAL OPERATIONS ON AN IMAGE

There are two different kinds of morphological operations:

1. Erosion
2. Dilation



Figure 1: Erosion Results

Notice the change in eyes, illuminates spots in the eyes are removed because in the input image there is a stark change in illumination at points near pupil. Dilation dilates the image. It tries to bring uniformity in image by converting dark points in neighborhood of points of higher intensity into bright ones.



Figure 2: Dilation Results

Dilation dilates the image. It tries to bring uniformity in image by converting dark points in neighborhood of points of higher intensity into bright ones.

THRESHOLDING AN IMAGE

Thresholding an image is one of the simplest ways of image segmentation. As the name suggests, it carries out its change according to a set threshold.

To threshold an image following function is used:

```
cvThreshold(input, output, threshold, maxValue, thresholdType)
```

Before we threshold the image we need to make a clone of the image.

IMAGE DATA

An image's data is stored as a character array whose first element is pointed by:-

```
Input->imageData (char pointer)
```

Number of array elements in 1 row is stored in

```
input->widthStep
```

Accessing pixel values in a grayscale image:

To find the pixel value in an image we need to define a pointer of type uchar:

```
uchar *pinput = (uchar*)input->imageData;
```

VIDEO INPUT

A video is basically a collection of continuous images displayed at a certain rate (generally 30 frames per second).

To extract the frames from video first we need to attach this video to the input stream and then extract those as and when required.

To attach the video to input stream we use the following function

```
CvCapture*
```

```
capture=cvCreateFileCapture("file_name.avi");
```

And for extracting frames use the following function:

```
IplImage*input_frame=cvQueryFrame(capture);
```

CAMERA INPUT

First camera needs to be initialized and then image is captured and further operations can be carried out on that image.

Use the following command for initiating the camera:

```
CvCapture
```

```
*capture=cvCreateCameraCapture(0);
```

0 stands for default webcam and to access other camera connected to the computer use 1 as argument.

Starting a camera takes time, so make sure that sufficient time has passed before we capture the image. This can be achieved through

```
for (int i=0; i<100000000&& capture! =NULL; i++);
```

Finally image is captured and stored in variable of type IplImage*

```
frame=cvQueryFrame(capture);
```

VIDEO INPUT THROUGH CAMERA

This is similar to video input all you need is attach the video from camera to the input stream. Following function helps us do so:

```
CvCapture
```

```
*capture=cvCreateCameraCapture(0);
```

There is no need to initialize the camera in this case because frame is captured regularly. Again, 0 for default webcam and use 1 for input through external camera.

HAAR CASCADES

Haar like features are digital image features used in object recognition. Haar Cascades are trained classifiers used for detecting features like face, eyes, upper body etc.

Firstly, you need to load cascade and then use the cascade to detect the presence of the corresponding feature. In most cases you need to mark the region of your interest.

BLOB DETECTION EDGE DETECTION

Here is the simple algorithm going to count the number of pixels in the blob. This could easily be extended to actually collect the pixel locations to create a region.

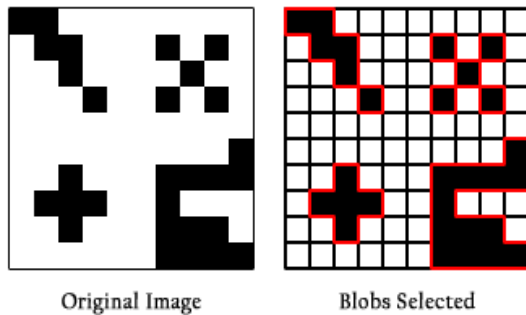


Figure: 3 Edge detection

In this example black pixels are “on” and white pixels are “off”.

CYGWIN

A large collection of GNU and Open Source tools which provide functionality similar to a Linux distribution on Windows. A DLL (cygwin1.dll) which provides substantial POSIX API functionality.

INSTALLING A C++ COMPILER, DEBUGGER, AND MAKE ON WINDOWS

The first step is to get the following components installed on your system:

- g++, the GNU C++ compiler,
- gdb, the GNU debugger, and
- make, a utility for compiling and linking multi-file projects.

**SYSTEM ANALYSIS AND DESIGN
CROWD DENSITY ESTIMATION MODEL**

Crowd density is classified into five distinct levels based on the five levels of service as shown in Table I. Each level of service is defined based on the range of average area occupancy for a single pedestrian. The five density groups are very low (VL), low (L), moderate (M), high (H), and very high (VH). Pixel-based methods give us reliable results when crowd density is low. And texture-based methods have outstanding performance when the crowd density is high. In order to achieve good estimation results, we need to develop a solution to solve the adaptive crowd density estimation for crowds real-time monitoring.

Table I Level of service

Level of Service	Density Range(people/m ²)	Group
A: Free flow	<0.50	Very Low
B: Restricted flow	0.50-0.80	Low
C: Dense flow	0.81-1.26	Moderate
D: Very dense flow	1.27-2.00	High
E: Jammed	>2.00	Very High

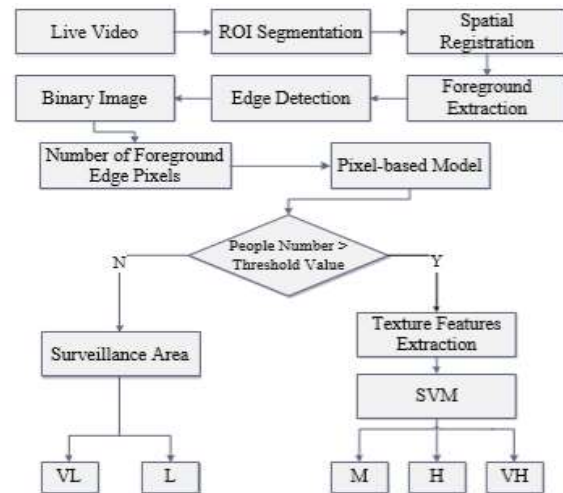


Figure 4: Existing crowd density estimation Model

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.

Execution Time: 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.

Test of Stereoscopic Measures: 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.

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

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 diagram.

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.

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.

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.

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.

IMPLEMENTATION

The four major modules that are defined as follows:

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. 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. 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.

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. It should be noted that these images constitute the proposed algorithm's data input.

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.

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. 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.

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. 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, 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 $\mu = 5\text{cm}$ and $\sigma = 10\text{cm}$. In this paper, we used a canny edge detector. This detector presented the best performance and robustness, in addition to obtaining all the edges in terms of 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. 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_1, y_1) , called the template, is selected from the current rectified image1. 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 image2, an area is selected in which the correspondence may be located.

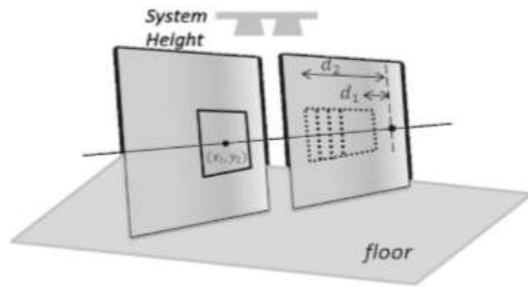


Figure 5: Matching the possible disparity values

As reported, the depth value in our specific configuration is calculated according to below equation, as explained in the section on image rectification

$$\text{depth} = f \cdot B / dx \quad - (1)$$

Where f is the focal distance common to the system, B is the distance between the cameras' optical centers (baseline), and dx 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.

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. Degeneration problems occur in the particle filter, and thus, to reduce this effect, a resampling step is included. 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 (probability density function).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

http:// www.ijesrt.com

independent identical distribution (i.i.d). All the particle weights $w^{(i)}$ 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.

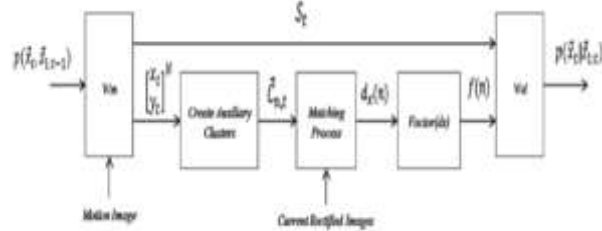


Figure 6: Observation stage in the extended particle filter with random reinitialization.

The weight associated with each particle is maintained equal to its weight prior to predicting its state vector.

Observation: Each of these factors has a fixed percentage of influence on the weight function, determined by α and β . 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 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 $\beta = 0.4$.

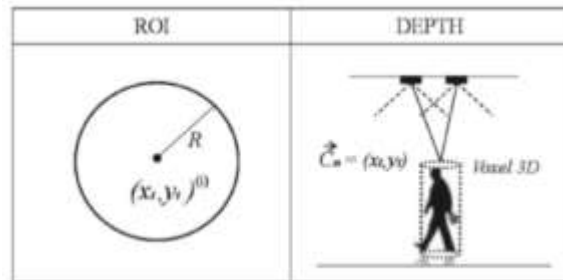


Figure 7: Factors affecting the weight of each particle: Region-based optical flow and depth.

1) **Selection:** For the selection process, a uniform random multinomial resampling of the discrete belief is carried out, which provides the correction stage. It is necessary to normalize all particle weights $w(i)$. In this way, more of the particles with greater weight are regenerated than particles presenting a lower weight.

Clustering Method

This process is based on a standard K-means algorithm 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 are not known a priori. Therefore, we incorporated a modification into the algorithm, we eliminated the need to give a prior 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 corresponding cluster.

Tracking and Counting People

To perform a count, it is necessary to construct trajectories using the detections 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. The calculation of the distances between trajectories and detections, in the NN, is carried out using the Euclidean distance of the characteristics that form detection. Inputs and outputs are counted according to the start and the finish point of a tracking. The cross lines include various articles 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.

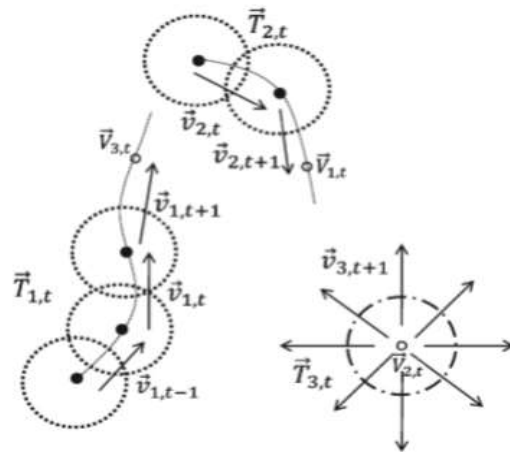


Figure 8: Proposed hypothesis tracking.

RESULT ANALYSIS

Pre-processing and segmentation of video into images :

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. 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.

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.

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.

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.

a. Input Image after segmentation from Video into frames:



Figure 9: Input Image after segmentation

b. Image Preprocessing results :

Head detection results after Morphological operations



Figure 10: Image Preprocessing

c. Tracking People with CVrect function :



Figure 11: Tracking people

d. Actual count per frame in a video :



Figure 12: Real counting of persons.

CONCLUSION

In this phase, 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. Stereo vision 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.

FUTURE WORK

The Map Reduce platform has been widely used for large-scale data processing and analysis recently. It works well if the hardware of a cluster is well configured. However, our survey has indicated that common hardware configurations in small and medium-size enterprises may not be suitable for such tasks. This situation is more challenging for memory-constrained systems, in which the memory is a bottleneck resource compared with the CPU power and thus does not meet the needs of large-scale data processing. The traditional high performance

computing (HPC) system is an example of the memory-constrained system according to our survey. We plan to optimize the performance for running multiple jobs simultaneously our proposed model. In order to realize the multi-job mode, i.e., run multiple jobs simultaneously, the same methodology as in this paper (i.e., the global memory and I/O management) can be applied. In order to optimize the performance in the multi-job mode, the key additional consideration is to take into account each job's characteristics. We plan to explore the following two potential approaches to achieving in future analysis.

REFERENCES

- [1] PARAGIOS N, RAMESH V. A MRF Based Approach for Real-Time Subway Monitoring[C] Proceedings of 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition: December 8-14, 2001. Kauai, HI, USA, 2001: 1034-1040.
- [2] LIN S F, CHEN J Y, CHAO H X. Estimation of Number of People in Crowded Scenes Using Perspective Transformation [J]. IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans, 2001, 31(6): 645-654.
- [3] SUBBURAMAN V B, DESCAMPS A, CARIN-COTTE C. Counting People in the Crowd Using a Generic Head Detector[C] Proceedings of 2012 IEEE 9th International Conference on Advanced Video and Signal-Based Surveillance (AVSS): September 18-21, 2012. Beijing, China, 2012: 470-475.
- [4] WU Xinyu, LIANG Guoyuan, LEE K K, et al. Crowd Density Estimation Using Texture Analysis and Learning[C] Proceedings of IEEE International Conference on Robotics and Biomimetics, 2006 (ROBIO'06): December 17-20, 2006. Kunming, China, 2006: 214-219.
- [5] DOLLAR P, TU Zhuowen, PERONA P, et al. Integral Channel Features[C] Proceedings of 2009 British Machine Vision Conference: September 7-10, 2009. Rama Chellappa, UK, 2009: 1-11.
- [6] BROSTOW G J, CIPOLLA R. Unsupervised Bayesian Detection of Independent Motion in Crowds[C] Proceedings of 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06): June 17-22, 2006. New York, NY, USA, 2006: 594-601.
- [7] CONTE D, FOGGIA P, PERCANNELLA G, et al. A Method for Counting People in Crowded Scenes[C] Proceedings of 2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS): August 29-September 1, 2010. Boston, MA, USA, 2010:
- [8] 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.
- [9] 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.
- [10] 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.
- [11] 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.
- [12] 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.
- [13] 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.
- [14] 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.
- [15] 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.
- [16] S. Yu, X. Chen, W. Sun, and D. Xie, "A robust method for detecting and counting people," in Proc. ICALIP, 2008, pp. 1545-1549.
- [17] 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.
- [18] B. Karthik, T.V.U. Kiran Kumar, M.A. Dorairangaswamy and E. Logashanmugam, Removal of High Density Salt and Pepper Noise Through Modified Cascaded Filter, Middle-East Journal of Scientific Research 20 (10): 1222-1228, 2014, ISSN 1990-9233.
- [19] S.P. Vijayaragavan, B. Karthik and T.V.U. Kiran Kumar, Privacy Conscious Screening Framework for Frequently Moving Objects, Middle-East Journal of Scientific Research 20 (8): 1000-1005, 2014, ISSN 1990-9233.