People Counting in Extremely Dense Crowd using Blob Size Optimization

Muhammad Arif[1,2], Sultan Daud[1,2], Saleh Basalamah[1,2]

[1]Center of Research Excellence in Hajj and Omrah (HajjCoRE), Umm Al-Qura University, Makkah, Saudi Arabia
[2]College of Computer and Information Systems, Umm Al-Qura University, Makkah, Saudi Arabia
syedmarif2003@yahoo.com

Abstract: Estimating Crowd density and counting people is an important factor in crowd management. The increase of number of people in small areas may create problems like physical injury and fatalities. Hence early detection of the crowd can avoid these problems. Counting of the people moving in the crowd can provide information about the blockage at some point or even stampede. In this paper, we have proposed a framework to count people in the extremely dense crowd where people are moving at different speeds. Foreground segmentation is done by various methods of background subtraction namely, approximate median, and frame difference and mixture of Gaussian method. Time complexity is calculated for these techniques and approximate median technique is selected which fast and accurate. Blob analysis is done to count the people in the crowd and blob area is optimized to get the best counting accuracy. Proposed framework is analyzed for three videos from Al-Haram mosque and people counting accuracy is found to be more than 96% in all three videos**.**

[Arif M, Daud S, Basalamah S. **People Counting in Extremely Dense Crowd using Blob Size Optimization**. *Life Sci J* 2012;9(3):1663-1673] (ISSN:1097-8135). http://www.lifesciencesite.com. 242

Keywords: People Counting; Extremely dense crowd; Blob Analysis; Foreground Segmentation

1. Introduction

As the population of the world is growing day by day, maintaining the public order in the crowded areas of the big cities is getting very important. Some examples are Airports, railway stations, carnivals, concerts and sports events. Extensive use of closed circuit monitoring system is in place in major cities. Moreover, estimation of crowd or number of people attending certain event is also becoming important for government agencies, public opinion making and news channels. Hence lot of research is being done in automating the process of estimation and management of crowd using visual cameras, thermal imaging or other sensors placed at the entry points. Crowd management during Ramadan and Hajj at the Holy Mosque in Makkah is a daunting task. Tremendous effort from the security staff is required to manage the huge crowd peacefully and smoothly. In the last decade, due to low cost of cameras, lots of cameras were used for the surveillance of public places.

Manual monitoring of crowd is done by putting many surveillance cameras and some observers monitor the crowd density and their movement. In this scenario, the cost of surveillance is very high. Alertness of the observers is an important factor in good surveillance. As the working hours increases as the case of Masjid-e-Haram, fatigue and stress of the observer increases degrading their performance. The importance and demand for automated tools to manage and analyze crowd behavior and dynamics grows day by day as the population increases. Major works started in early 1990s as researchers employed various technologies and techniques to come up with different solutions to problems.

Most of the research in the field of crowd density estimation has focused on either segmentation of people or head counts, or based on texture analysis or wavelet descriptors. Some research is related to removing the background area from the foreground and then crowd density is estimated based on the foreground area assumed to be occupied by the people.

(Zhao et al 2003) proposed Bayesian model based segmentation to segment and count people but this method is not appropriate for high density crowds. (Yoshinaga et al 2010) proposed blob features of moving objects to eliminate background and shadow from the image. For each blob of moving people, numbers of pedestrians are estimated by using neural networks. They have shown that accuracy of 80% can be achieved by this method in the real life scenarios where maximum numbers of pedestrians are 30 in a single frame.

(Xiaohua et al 2005) showed classification accuracy of 95% when crowd density is classified into four classes by using wavelet descriptors. Classification is done by support vector machine. Their method is good for estimation of crowd density for moderate crowd density.

(Ma et al 2008) used texture descriptors called advanced local binary pattern descriptors to estimate crowd density estimation. They have calculated LBP from the blocks of the image and tested on small database of images for automatic surveillance. They divide the image into squares, bottom squares are bigger, upper ones are half the size. The ground truth is manually labeled for each square, low for 0-0.5 persons in a square etc. with 5 categories in total. They classify the images cells (squares) using kmeans clusters and the distance is computed using their pattern descriptor. Once the training of cells is done and clusters are formed, they test data for each cell, they find the texture and put that in a group, hence adding all the squares they get crowd estimation. This is yet another example of using texture. May not be suitable for generic apps, but it seems to work well once trained.

Terada et al. proposed a system that calculate the directional movement of the crowd and count the people as they cross some virtual line (Terada et al 1999). (Hashimoto et al 1997) used specialized imaging system using infra-red imaging to count the people in the crowd. (Davies et al 1995) have discussed in detail the concept of crowd monitoring using image processing through visual cameras. (Roqueiro et al 2007) used simple background subtraction from the static images to estimate the crowd density.

Some other researchers (Velastin et al 1993, 1994) have also used the concept of background removal to estimate the crowd area. Computer vision algorithms were employed to monitor crowd densities and behaviors with various degrees of accuracies. (Velastin et al 1993, 1994) dealt with crowd densities and count, and motion estimation. They fixed an area to observe, and then asked people to walk past it normally in different number, they have the background image (empty), then they counted the people manually in each image, and got the background subtracted image, and the thin edge images of people, they plotted the number of people vs. positive pixels and make a graph. Using a kalman filter they combined the line thinning and blob methods and ran automated trials. This took care of the density vs. the people count in a confined area. To measure motion they used optic flow, they assumed that if the motion stops something has happened, so we get the people present from the BG subtracted image, and flow from optic flow. They tend to use a flow smoothness technique for the objects that move more than one pixel between processed frames.

(Reisman et al 2004) used a forward facing camera mounted on the car to detect crowd of pedestrians. It assumes that a camera in a moving forward car will have outward optic flow. Any moving objects will produce inward optic flow hence they detect the motion. They also use classifiers to distinguish between human and cars. They use a variant of Hough line transform to detect the disturbances in the optic flow due to moving objects. The crowd is not assumed to be traveling in a particular direction and hence if moving in a haphazard manner, optic flow cannot help in this case.

To estimate the crowd density using image processing, many researchers have used the information of texture, edges or some global or local features (Marana et al 1997, Ma et al 2004, Lin et al 2001). (Marana et al 1997) argued that low density crowd images have course texture and high density crowd images have fine texture, they computed the texture using Gray Level Dependence Matrix at four angles computing contrast, homogeneity, energy and entropy to form a 16 parameter vector, and then trained SOM neural network on the densities and the vector relationships. The results were not too impressive in the low density scenarios, but decent in high density. This was supervised learning so different for each scenario/camera view.

(Ma et al 2004) argues that the perspective distortions in images for pixel based crowd estimation are either incorrect or not done well, they propose a geometric correction technique, and they argue that the correction depends on y-axis only. Hence if a human is standing upright, pixels on his feet have a scale, and all the pixels on his body has the same scale as his distance from the camera is same. They use a simple foreground pixel detection technique using some masks and adaptive area growing as well. They integrate the GC into their pixel count using a lookup table. They assume each person as a rectangle changing in size with y value, and then consider all positive pixels in that rectangle as that person. The authors point out many flaws in past research works but this approach may fail when dealing with high crown density when people occlude others partially and completely

Another work (Lin et al 2001) trained support vector machines using HAAR transform to identify heads of people after histogram equalization to eliminate illumination changes in a crowd in order to count them and estimate the densities. It used 16x16 pixel head templates and resized the image to get heads of various sizes. In Sheng's paper, none of the persons are wearing anything on their heads, in Masjid-e-Haram people may wear caps, and hijabs and this head classifier will fail.

Some researchers (Marana et al 1998a, 1998b, Ma et al 2008) have used texture analysis to extract certain features from the images and have used neural networks to estimate the crowd. (Cho et al 1998, 1999) and (Huang et al 2002) blended the concept of image processing and neural networks to estimate and count the crowd of people.

(Yang et al 2003) have used group of image sensors to segment the foreground objects from background scene to count the approximate number of people in the crowd in a particular scene. (Xiaohua et al 2006) have used wavelets to extract the features from the images for the crowd estimation

(Rodueiro et al 2007) uses the fore ground pixels and finds them using a Median Background computing technique. Foreground pixels are found by applying a threshold and then morphological operations are done to smooth the results. They ignore zones by masking area that have motion but not interesting like road (cars) etc. They apply classification algorithms like SVM, k-nearest, PNN, BPNN to classify the images in 2 categories first, zero persons and one and more persons. On more than zero people's categories it again applies the classification techniques to find the number of people in the scene. They train these classifiers on 70% of the images and test them on the 30% of the remaining images. The median filters are applied on the sequence of image results to get rid of the spiky errors. Also they use assorted grid to see if the accuracy increases.

Recently many researchers have started work on counting people and crowd estimation using infra-red sensors as the cost of the sensors is going down and installation of these cameras are become affordable. Moreover, these cameras can be used in total darkness and the images obtained from these cameras are invariant to the different colors of cloths, and different level of illuminations. Andersson et al. [24] have used thermal infra-red sensors in the long wave infra-red band and visual cameras and proposed the concept of sensor fusion to predict the crowd behavior. (Teixeira et al 2007) proposed lightweight camera sensor nodes to count the people in the indoor environment based on motion histogram. Recently many infra-red sensors specifically designed for people counting are available in the market (25, 26).

Not many papers are published related to crowd estimation or people counting in Masid-e-Haram. (Hussain et al 2011) have proposed pixel based crowd density estimation system. They have used crowd foreground blobs to classify the crowd into five ranges from very low to very high using neural networks. (Jasy et al 2010) have proposed a generalized framework for crowd surveillance research in the context of crowd in Masjid-e-Haram. (Sarmady et al 2011) has proposed an interesting model for circular tawaf around Kaaba.

2. Material and Methods

Crowd density estimation is an important tool in good crowd management. Moving crowd may create situations where people may get fatal injuries. Early detection of crowd motion and people counts can avoid serious situations and fatalities. Therefore, crowd detection and estimation has been the area of interest of most of computer scientist and researchers. As a solution, automatic crowd detection and monitoring methods based on computer vision technology were proposed to overcome the weaknesses of manual and traditional surveillance systems. Figure 1 explains the methodology, video is streamed from the camera to the system and system processes the video frame by frame. From every frame, foreground is segmented which represents the moving objects in the frame. In our particular application, moving objects are human beings. Blob analysis is done on the foreground to find out the independent blobs of a particular size which later counted to find out the total number of people in the frame who are in motion.

A. Foreground segmentation

Identifying moving objects in video sequence is a fundamental and critical task in video surveillance, human detection and tracking, and gesture recognition in human-machine interface. Foreground segmentation is an important pre-processing step for detecting the moving objects from the video. Pixels in the current frame that deviate significantly from the background are considered to be moving objects. These foreground pixels are further processed for object localization and tracking. Frame difference technique is the simplest form of foreground segmentation based on the difference of two consecutive frames by applying a threshold value to decide between background pixel and foreground pixel (Gonzalez and Wood 1992). This technique is prone to change in the illumination conditions of the video. Many methods exist for foreground segmentation, each with different strengths and weaknesses in terms of performance and computational requirements. (Piccardi 2004) has provided a good survey by comparing different background subtraction methods. Many background extraction methods do not perform well in different lightening conditions. It is claimed in (stauffer and Grimson 1999) that if the background is not visible for most of the time then average or median filtering may fails to extract the background. Hence, Mixture of Gaussians method is robust in the case when background is multi-modal. (Stauffer et al 1999) proposed Mixture of Gaussian model which is stable and robust and good in non-stationary backgrounds. High computational complexity of these algorithms makes them not suitable for online video processing. Hence, we are interested in relatively faster and simple foreground segmentation techniques which can give us sufficient segmentation accuracy. Hence, we have analyzed the performance of the following two techniques in our particular application where high density crowd is moving in congestion.

- Frame Difference

- Approximate Median

Frame Difference:

Frame difference is the simplest form of background subtraction. The current frame is subtracted from the previous frame. If the resultant difference in the pixel values is greater than a threshold T_s, the pixel of the frame is considered to be part of the foreground as given below,

$$\left| f_i - f_{i-1} \right| > T_s \qquad (1)$$

Where f_i is the pixel value of the i^{th} frame in the video.

Approximate Median

Median filter is designed by buffering N number of frames and median of these frames are calculated and a threshold is applied to detect the background of the video. This method is very effective but many frames have to be stored to calculate the median frame. In median filtering, the previous N frames of video are stored in the buffer and the background frame is calculated as the median of buffered frames. Then the background frame is subtracted from the current frame to find out the foreground pixels.

The approximate median method (McFarlane and Schofield 1995) gives a good alternate solution to the buffering the N frames in the memory to calculate the median frame. The first frame is taken as the background frame and for the next coming frames; if the pixel value of the current frame is greater than the background pixel then the pixel value of the background image is incremented by 1. If the pixel value of the current frame is less than the background pixel value then the pixel value of the background pixel is decremented by 1. Hence it is assumed that after sufficient number of frames, the background image will converge to the true median image of the video.

B. Blob Area Optimization

Blobs are the connected regions in a binary image. Blob analysis process is aimed at detecting point and/or regions in the image that differ in properties like brightness or area etc. For blob detection, image is first converted to binary image. Then next step is finding the connected components in the binary image. We have used "***bwconncomp***" and "***regionprops***" functions of Image Processing toolbox of Matlab (34). Following are the steps for finding the connected components in the binary image.

1. Search the unlabeled pixel, *p*
2. Label all the pixels in the connected component containing *p* by using flood fill algorithm.
3. Repeat the step 1 and 2 until all the pixels are labeled.

After finding connected components in binary image, the next step is to measure the properties of each connected component (object) in a binary image. In this paper, we are interested in measuring the 'Area' of each connected components. Area is the number of pixels in the region. Each binary image has a lot of connected components of variable size. We are interested in finding those connected components having area greater than some specific value. Area of the connected component differs depending upon the distance of camera from the scene. If the distance between the camera and crowd is less, greater will be number of pixels in a connected component and hence greater will be the blob size of the object. Hence the first step in people counting is to decide the optimal area of connected component. For this purpose, we have used four initial frames whose ground truth is available. In the iterative approach, we change the area of the blob size and count the people. This count is then compared with the ground truth of the frame (actual number of people in the frame). For each frame, optimal area is found for which the people count error was minimum. The whole framework is shown in figure 4. For each initial frame, foreground segmentation is done and all connected components are calculated. In the next step, blob area is varied and blobs are counted whose area is greater than the particular blob area. This blob count is compared with the number of people in the frame. The error between the blob count and actual number of people in the frame is calculated for the whole range of blob area. Same procedure is done for all four initial frames and optimal blob area is found for which the counting error between the number of blobs and the actual people in the frames is minimized. This optimal blob area is used to count in the people in all the frames of the video. To assess the performance of the proposed framework, three videos in the mattaf area of Al-Haram Mosque is recorded where extremely dense crowd is doing tawaf of the Kaaba. In sample frame of VD1, there is a rich background, and many people are entering the tawaf area and leaving this area from different sides. Some people are standing and praying which are in relatively very slow motion while remaining on their place of prayer. In the tawaf area (middle area) many people are moving at different speed. At the outer circle they are moving fast whereas in the inner circle the motion is slow. Frame rate of the video is 50 frames per second and frame resolution is 1920 × 1080. The video contains more than 10,000 frames. Since there are more than 2500 people in motion in every frame so it is extremely difficult to count in all the 10,000 frames. Moreover, in one second there are 50 frames and we expect that not much change occurs from one frame to the other frame. Hence we have decided the count the people in motion in 100 frames at almost equal interval of about two seconds. We assume that the error in the people counting occurring in these 100 frames will be the same in all 10,000 frames.

C. Description of Video for the testing of the proposed framework

In video VD2, there is a close up look of the people doing tawaf near Kaaba. Some people in green dress are cleaning the place, some security staff is there controlling the crowd and people are moving around in a very dense fashion. This video consists of 76 frames and ground truth of 12 frames is calculated out of which four frames will be used to optimize the parameters and eight frames will be used to assess the performance of the proposed framework.

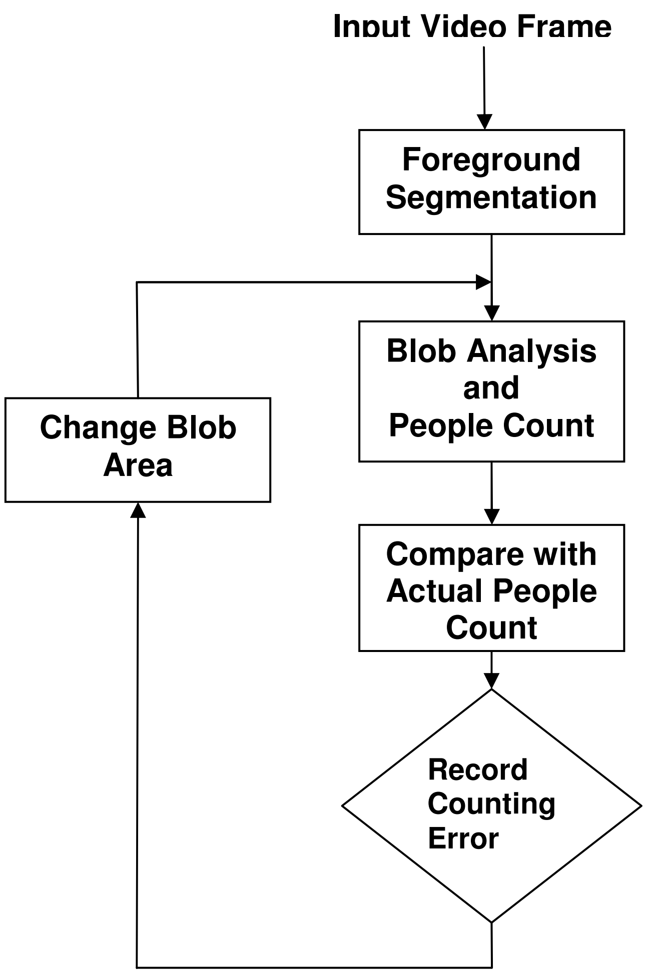


Figure 2: Blob Area estimation

Video VD3 is again a close up look of people doing tawaf by a different angle of camera. The crowd is again extremely dense. In this video most of the people are wearing Ehram (Two pieces of cloths wrapped around the body with one shoulder naked). This is a typical scenario during umrah and hajj. This video also contains 76 frames out of which ground truth of 13 frames are calculated to test the accuracy of people counting. So all three videos carry different setting and different scenarios and are good for testing the people counting algorithms.

3. Results

Foreground segmentation is done with different level of threshold values using approximate median. Threshold effect on foreground segmentation is very similar to the frame difference method. For low threshold values, it is difficult to differentiate among people in the closed vicinity and for very high threshold value, some people will be skipped. But one very important thing that can be observed in the figure that by varying the threshold value, a proper threshold value can be selected which can generate blobs distinguishing the people in the closed vicinity. By visual inspection of four initial frames it is predicted that optimal threshold value lies between 40 and 60 approximately. Foreground segmentation by approximate median at threshold level of 50 is shown in Figure 3.

A. Timing Analysis of Different Foreground segmentation methods

To study the time complexity of the foreground estimation methods described above, frame difference, approximate median and mixture of Guassian methods are used to extract the foreground frame for 100 frames of the video and the time is recorded as average frame processing time and is recorded in Table 1.

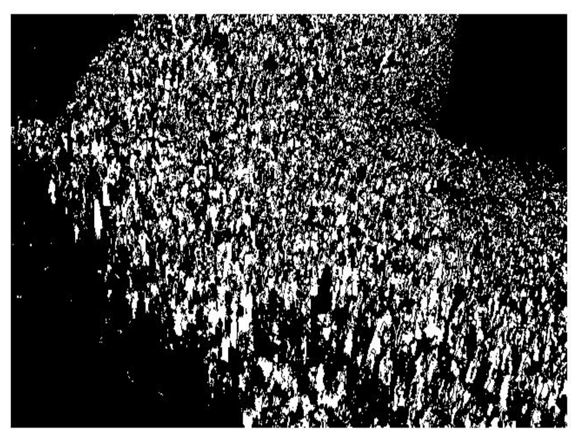


Figure 3: Foreground segmentation by approximate median method

The system used for calculating the time complexity is HP Compaq 8100, Processor is Intel(R) Core(TM) i5 CPU 2.80GHz, with 4GB RAM. All calculations are done in the Matlab 2011a environment. It can be seen that frame difference method is fastest and approximate median method takes almost double frame processing time. Mixture of Gaussian method is very expensive and is out of question for online video processing. Hence, we will compare frame difference method and approximate median method for the foreground extraction in terms of people counting accuracy.

Table 1: Time Complexity of Foreground Extraction Methods

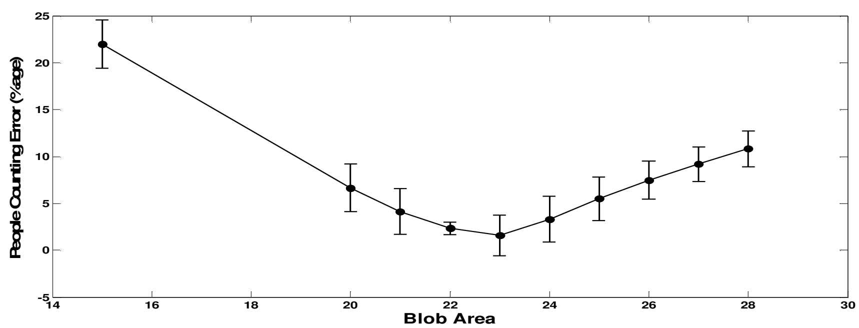
	Average Frame Processing Time (Seconds) Mean ± Standard deviation
Approximate Median	**0.249 ± 0.013**
Frame Difference	**0.137 ± 0.008**
Mixture of Gaussians	**62.37 ± 2.93**

After foreground segmentation, the next step is finding the optimal area of blobs. As described in section A, the optimal threshold value lies between 40 and 60, so the threshold value is selected to be 50 to find the optimal value of the blob area.

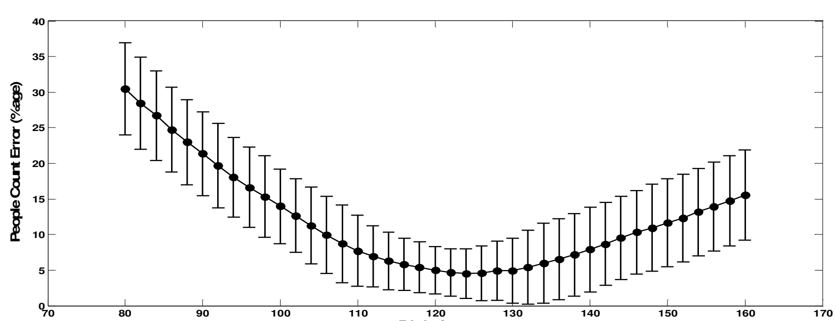
A range of bob area is defined manually and counting error is calculated by varying the blob area. Average and standard deviation of the error between the people count using the blob area and the actual number of people in all four frames is plotted in figure 4 versus the blob area. In Figure 4(a), mean and standard deviation of the counting error is plotted for VD1. It can be observed from the figure that the error is minimum for the blob area equals to 23. Hence blob area of 23 is taken to calculate the number of people in all the frames of VD1. Figure 4(b) shows the plot of error versus blob area for the video VD2. The best blob area is found to be 125 for which the error is minimized. Since the size of people is this video is larger as compared to VD1, the optimal blob area is also large. Counting error for VD3 is plotted in figure 4(c) and optimal blob area is found to be 110.

B. Optimization of Threshold Value

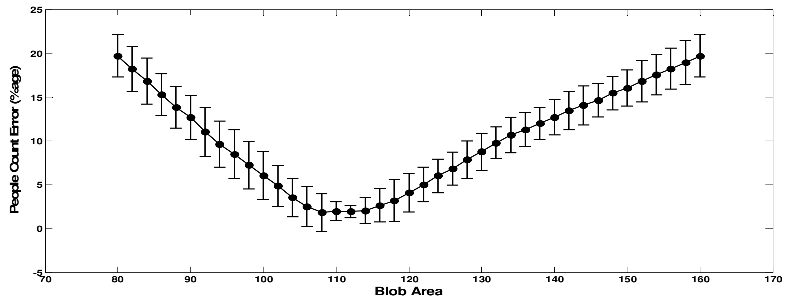
It is very important to select the proper value of the threshold so that foreground segmentation can be achieved in the optimal way. For this purpose, initial four frames are selected with the actual people count in the frames from VD1. Blob area as discussed in section B, is selected as 23 for VD1. For every threshold value, people are counted and compared with the ground truth (Actual number of people moving in the frame). Average error in the people count is plotted in Figure 5 with the standard deviation as function of threshold value for approximate median. Optimal threshold value is found to be 50 on which we have got the minimum error.



(a) Video VD1



(b) Video VD2



(c) Video VD3

Figure 4: Optimization of blob area for four different frames for videos VD1, VD2, VD3

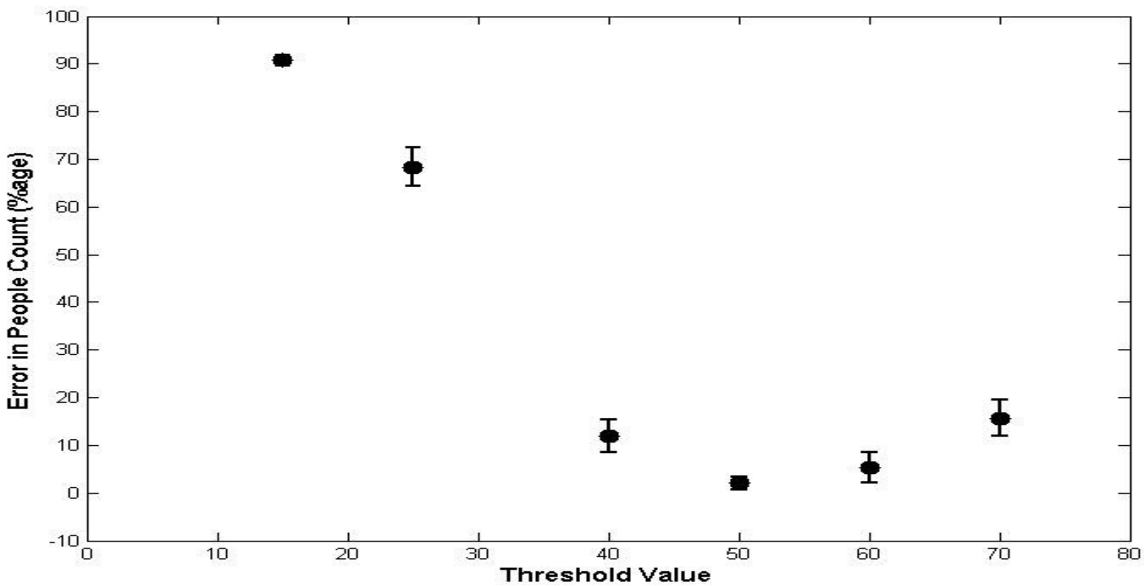


Figure 5: Average counting error for different Threshold values using approximate median method

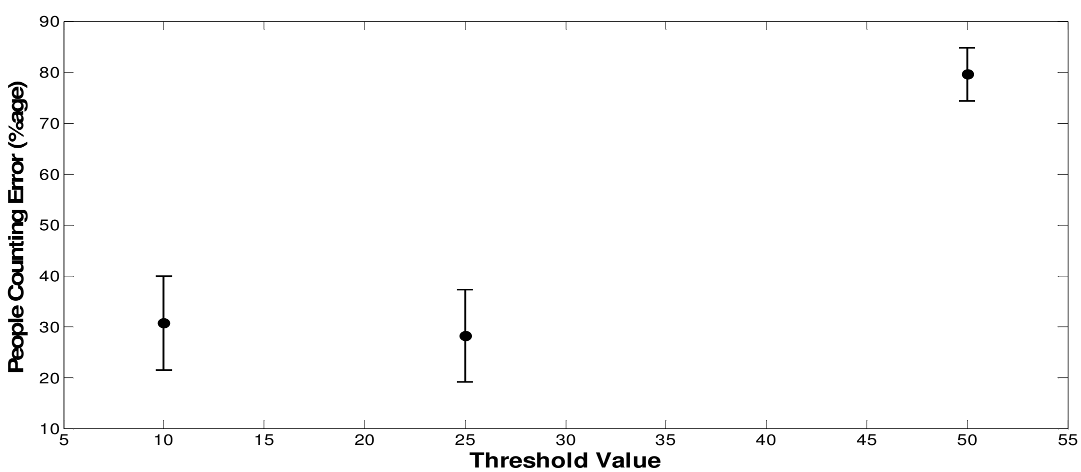


Figure 6: Average counting error for different Threshold values using frame difference method

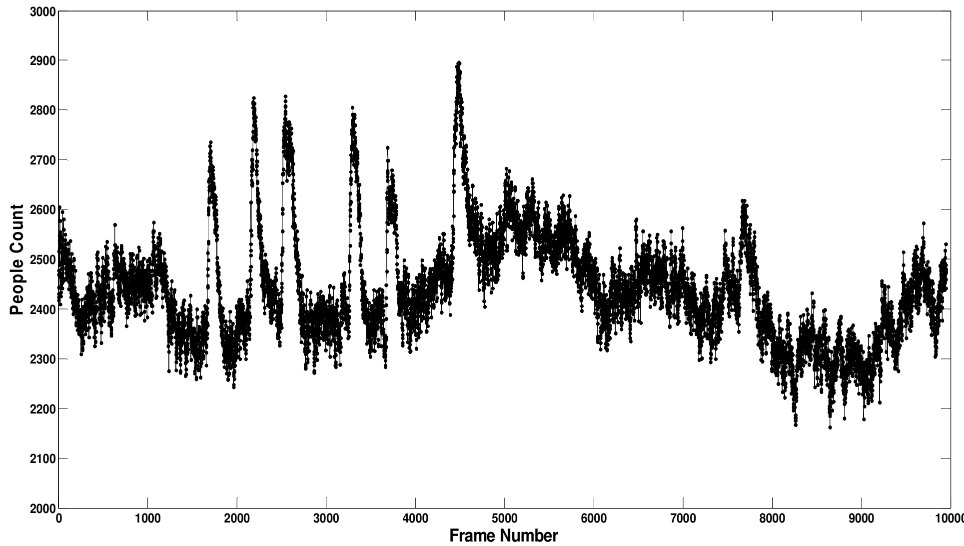


Figure 7: People count for all the frames in VD1

Similarly, in figure 6, average of counting error with standard deviation is plotted for the frame difference method. For the frame difference method, the optimal threshold value is found to be 15 at which the counting error has minimum value.

But comparing figure 5 and 6, it can be seen that approximate median is far better method as compared to the frame difference method in terms of counting accuracy. Hence we have selected approximate median method for the foreground segmentation with the threshold value equals to 50.

C. Performance analysis of the proposed framework

For the fixed setting of threshold value for foreground segmentation (equals to 50) and blob area of 23, people are counted in all the frames of VD1 (11,000 frames). To check the counting accuracy of the proposed framework, ground truth (actual number of people moving in the frame) is calculated for the frames after equal intervals. In figure 7, people count for 10,000 frames are plotted and it can be observed from the figure that people are coming and leaving the tawaf area. Total number of people remains between 2200 and 2900.

To check the accuracy of the counting of people, counting error is calculated and plotted in Figure 8 for 92 frames. Average counting error is found to be 3.5% with standard deviation of 3.1% which is very good in the scenario where the crowd is extremely dense, and people are moving at different speed in the video.

Similarly, number of people in video 2 is also plotted in Figure 9 for about 70 frames. The optimal blob area equals to 125 is used to count the people. The range of people count is between 20 and 640 approximately. Accuracy of the method is checked for eight frames selected at approximately equal time durations. The data is tabulated in Table 2 for the eight frames. Average people counting error is found to be 3.6%. It is assumed that same counting error will exist in the counting of all 70 frames.

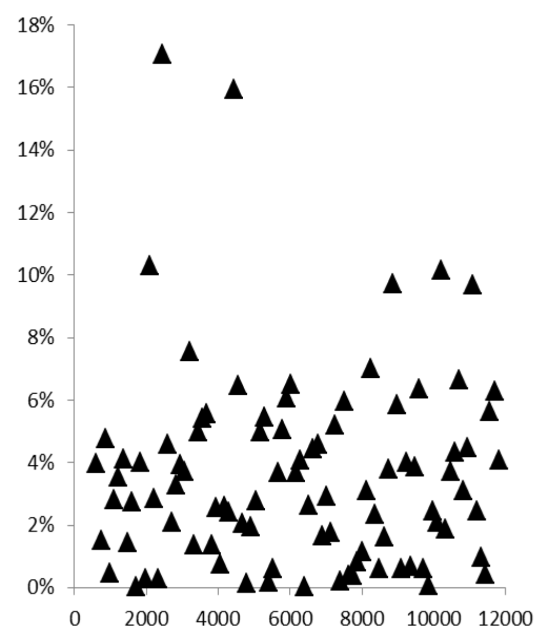


Figure 8: People count error (in percenatge) for selected frames in VD1. Overall Error is **3.5% ± 3.1%** (Error on y-axis and frame number on x-axis)

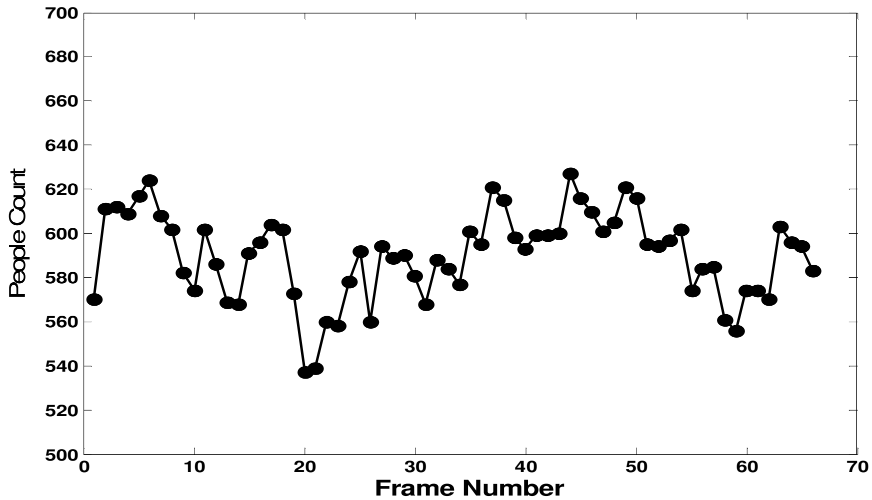


Figure 9: People count for VD2

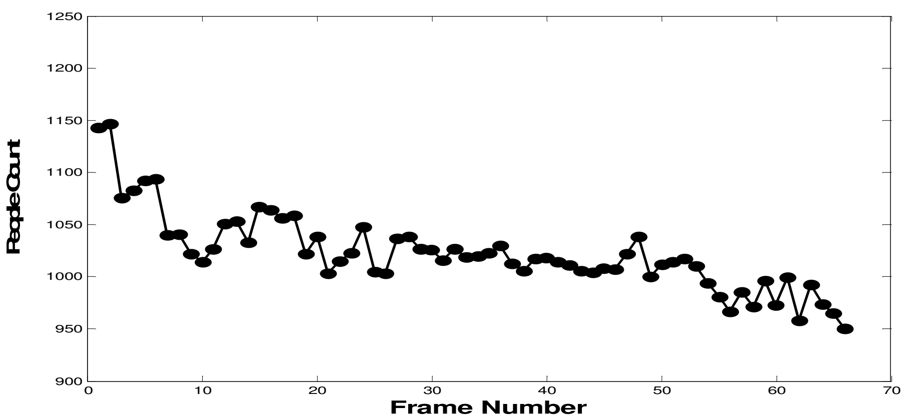


Figure 10: People count for VD3

Table 2: People Counting Error in VD2

Frame Number	Ground Truth	People Count	Error
15	595	617	3.69%
20	598	574	4.01%
30	592	537	9.29%
40	594	581	2.18%
45	595	601	1.00%
55	590	616	4.40%
60	589	616	4.58%
65	593	574	3.20%
75	595	594	0.16%
		Average Error	3.62%

In VD3, people are counted with the optimal blob area of 110 and threshold value of 50 for foreground segmentation. People count is plotted in Figure 10 for all 76 frames. It can be seen from the figure that number of people in the video is gradually decreased from 1150 to 950. Again accuracy of the method is checked for ten frames which are at almost equal duration of time and tabulated in Table 3. The average counting error is found to be 2.6% only.

A good accuracy of more than 96% is observed in all three videos which show the effectiveness of the proposed framework. It can be applied to the different focus settings of the camera just by optimizing the threshold and blob area values. In all three videos we have assumed that camera is watching the crowd from approximately top position and hence the effect of angled view is not significant.

Table 3: People Counting Error in VD3

Frame Number	Ground Truth	People Count	Error
18	1052	1041	1.04%
25	1031	1067	3.49%
30	1021	1038	1.66%
36	1021	1003	1.76%
42	1011	1027	1.58%
48	1014	1006	0.78%
55	977	1008	3.17%
60	1014	1012	0.19%
65	1020	981	3.82%
70	1030	973	5.53%
Average Error			2.31%

In this paper we have considered extremely dense crowd and proposed a framework to count the people moving in the video in this crowd with different speed. A test case of Al-Haram mosque is considered in which hundreds of people are doing tawaf (circling around) of Kaaba. Threshold value is optimized for the foreground segmentation and timing analysis was done to find out its suitability for the online video processing. Blob area optimization is done for every video to find out the appropriate blob area for a particular setting of the camera. It is observed that proposed framework worked very well in counting the moving people in the extremely dense crowd with counting accuracy of more than 96% in all three videos. This validates the efficacy of the proposed framework in counting the extremely dense crowd.

Acknowledgements:

This research has been supported by the Center of Research Excellence in Hajj and Omrah (HajjCoRE), Umm Al-Qura University. Under Project number P1119, titled "Automatic Decision Support System for Crowd Estimation and Management in Masjid-e-Haram ". Authors also acknowledge the help of Mr Omer Ishaq and Ms Naima Fahad, Air University in this research.

Corresponding Author:
Dr. Muhammad Arif
Department of Computer Science
Abdiyah Campus, Umm Alqura University
Makkah, Saudi Arabia
E-mail: syedmarif2003@yahoo.com

References

1. Zhao T, Nevatia R. Bayesian human segmentation in crowded situations. IEEE Conference on Computer Vision and Pattern Recognition, 2003;2:495-466.
2. Yoshinaga S, Shimada A, Taniguchi R. Real-time people counting using blob descriptor, Procedia Social and Behavioral Sciences, 2010;2:143-152.
3. Xiaohua L, Lansun S, Huanqin L. Estimation of crowd density based on wavelet and support vector machines. International Conference on Intelligent Computing, 2005.
4. Ma W, Huang L, Liu C. Advanced local binary pattern descriptors for crowd estimation, IEEE Pacific Asia Workshop on Computational Intelligence and Industrial application, 2008:958-962.
5. Terada K, Yoshida D, Oe S, Yamaguchi J. A method of counting the passing people by using the stereo images. International conference on image processing, 1999.;2:338-342.
6. Hashimoto K, Morinaka K., Yoshiike N, Kawaguchi C, Matsueda S. People count system using multi-sensing application. International conference on solid state sensors and actuators, 1997;2:1291-1294.
7. Davies AC, Yin JH, Velastin SA. Crowd monitoring using image processing. IEE Electronic and Communications Engineering Journal, 1995;7(1):37-47.
8. Roqueiro D, Petrushin VA. Counting people using video cameras. International Journal of Parallel, Emergent and Distributed Systems (IJPEDS), 2007;22(3):193-209.
9. Velastin SA, Yin JH, Davies AC, Vicencio-Silva MA, Allsop RE, Penn A. Analysis of crowd movements and densities in built-up environments using image processing. Proc. IEE Colloquium Image Processing for Transport Applications, 1993;8:1-6.
10. Velastin SA, Yin JH, Davies AC, Vicencio-Silva MA, Allsop RE, Penn A. Automated measurement of crowd density and motion using image processing, in Proc. 7th Int. Conf. Road Traffic Monitoring and Control, 1994:127-132.
11. Reisman P, Mano O, Avidan S, Shashua A. Crowd detection in video sequences, International Symposium on Intelligent Vehicles, 2004:66-71.
12. Marana A, Velastin SA, Costa L, otufo R. Estimation of crowd density using image processing. Image Processing for Security Applications, IEE Colloquium, 1997;11:1-8.
13. Ma, R Li L, Huang W, Tian Q. On pixel count based crowd density estimation for visual surveillance. IEEE Conference on

Cybernetics and Intelligent Systems , 2004;1:170-173.
14. Lin SF, Chen JY, Chao HX. Estimation of number of people in crowded scenes using perspective transformation, IEEE Transactions on Systems, Man and Cybernetics, 2001;31(6):645-654.
15. Marana AN, Costa LF, Lotufo RA, Velastin SA, On the efficacy of texture analysis for crowd monitoring. Proc. Computer Graphics, Image Processing, and Vision, 1998:354-361.
16. Marana AN, Velastin SA, Costa LF, Lotufo RA, Automatic estimation of crowd density using texture. Safety Science, 1998;28:165-175.
17. Ma W, Huang L, Liu C. Advanced Local Binary Pattern Descriptors for Crowd Estimation, IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application. 2008;2:958-962.
18. Cho Y, Chow TWS. A fast neural learning vision system for crowd estimation at underground stations platform, Neural processing letters, 1999;10(2):111-120.
19. Cho Y, Chow TWS, Leung CT. A neural based crowd estimation system by hybrid global learning algorithm, IEEE Trans on Systems, Man, and Cybernetics, Part B, 1999;29(4):535-541.
20. Huang D, Chow TWS, Chau WN. Neural network based system for counting people, IEEE Trans on Systems, Man, and Cybernetics, Part B, 2002;31(4): 2197-2200.
21. Yang DB, Gonzalez-Banos HH, Guibas LJ. Counting people in crowds with a real-time network of simple image sensors, Proceedings of Ninth IEEE International Conference on Computer Vision, 2003;1:122-129.
22. Xiaohua L, Lansun S, Huanqin L. Estimation of Crowd Density Based on Wavelet and Support Vector Machine. Transactions of the Institute of Measurement and Control, 2006:299-308.
23. Roqueiro D, Petrushin VA, Counting people using video camera, International Journal of Parallel, Emergent and Distributed Systems , 2007;22(3).
24. Teixeira T, Savvides A, Lightweight People Counting and Localizing in Indoor Spaces Using Camera Sensor Nodes, First ACM/IEEE International Conference on Distributed Smart Cameras, 2007:36-43.
25. Thermal Imaging People Counters, http://www.sensourceinc.com/thermal-video-imaging-people-counters.htm
26. IRISYS - InfraRed Integrated Systems, http://www.irisys.co.uk/people-counting/our-products/
27. Hussain N, Yatim HSM, Hussain NL, Yan JLS, Haron YF. CDES: A pixel-based crowd density estimation system for Masjid al-Haram, Safety Science, doi:10.1016/j.ssci.2011.01.005, 2011.
28. Yan JLSa, Haron YF, A survey of crowd density estimation methods for high density crowds, Journal of Advanced Computing and Application 2010;1(1):8-25.
29. Sarmady S, Haron F, Talib AZ. A cellular automata model for circular movements of pedestrians during Tawaf. Simulation Modeling Practice and Theory, 2011;19:969-985.
30. Gonzalez RC, Woods RE. Digital Image Processing, Addison Wesley, 1st edition, 1992.
31. Piccardi, M. Background subtraction techniques: a review, IEEE International Conference on Systems Man and Cybernetics, 2004:3099-3104.
32. Stauffer C, Grimson WEL. Adaptive background mixture models for real-time tracking. In Proc. Computer Vision and Pattern Recognition, 1999;2:246-252.
33. McFarlane NJB, Schofield CP. Segmentation and tracking of piglets in images. Machine Vision and Applications, 1995;8(3):187-193.
34. MATLAB version 2011a. Natick, Massachusetts: The MathWorks Inc., 2011.

7/30/2012