Counting of Moving People in the Video using Neural Network System

Muhammad Arif[1,2], Muhammad Saqib[1,2], Saleh Basalamah[1,2] and Asad Naeem[1,3]

[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
[3]Department of Computer Sciences, Air University, Islamabad, Pakistan
syedmarif2003@yahoo.com

Abstract: Automatic counting of people in the crowd using surveillance visual camera is very useful in effective crowd management, security surveillance, and many more applications. In this paper, we have proposed an intelligent framework to automate the process of people counting in the surveillance video. Foreground (moving people) segmentation from the video is done by combination of different foreground estimation techniques. Texture analysis and foreground pixel area for different segmentation techniques are used to extract the useful features. Neural Network is trained on these features and people counting accuracy of more than 96% is achieved on a benchmark video.

[Arif M Saqib M, Basalamah S, Naeem A. **Counting of Moving People in the Video using Neural Network System**. *Life Sci J* 2012;9(3):1384-1392] (ISSN:1097-8135). http://www.lifesciencesite.com. 201

Keywords: People Counting, Video Processing, Foreground segmentation, Texture analysis, Neural networks

1. Introduction

The importance and demand for automated tools to manage and analyze crowd behavior and dynamics grows day by day as the population increases. 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. Major works started in early 1990s as researchers employed various technologies and techniques to come up with different solutions to problems.

Major 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. (Velastin et al 1993, 1994) dealt with crowd densities and count. They fixed an area to observe, and then asked people to walk past it normally in different numbers, they had 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 used the area of pixels in the foreground image and tried to correlate it with the people count. (Zhao et al 2003) proposed Bayesian model based segmentation to segment and count people. (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. Few researchers have proposed model free people counting in the scene by statistical descriptors (Albiol et al 2009, Brostow and Cipolla 2006). (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 into classes from low density to high density. They classified the squares using k-means clusters and the distance is computed using their pattern descriptor. 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.

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. 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). (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.

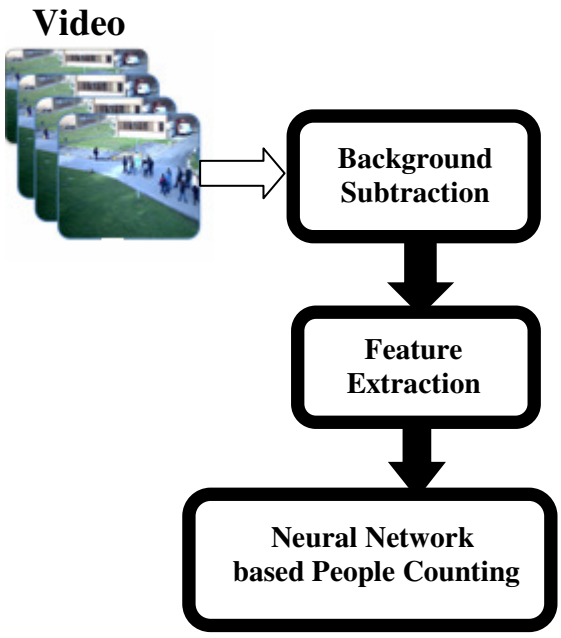


Figure 1: Proposed framework for the counting of people in the video

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. (Cho et al 1999) and (Huang et al 2002) blended the concept of image processing and neural networks to estimate and count the crowd of people. Crowd levels are detected in (Lo and Velastin 2001) and (Dan and Doug 2005) using classifiers. (Stauffer and Grimson 1999), Aguilera et al 2006) have used mixture of Gaussian model to adaptively extract the foreground containing the pixels in motion. (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.

2. Material and Methods

Figure 1 shows the block diagram of the overall people counting framework. In the first step background is modeled from some frames of the video and this background is subtracted from the video to segment the foreground of the video. In the foreground, it is assumed that only people are in motion. In the next step, texture analysis of the foreground frame is done to extract the features. These features are then fed to a trained neural network which output the number of people present in a particular frame. In the next sub-sections, we will discuss each block of figure 1 separately.

A. Background Extraction

Background modeling is an important and fundamental task in many computer vision applications. One very important application is to find moving objects using foreground segmentation. In foreground segmentation, background model is subtracted from each and every frame to find the moving foreground objects in the video. There are many challenges associated with background modeling. Good background model must adapt to the changing conditions such as illumination, shadowing, and other non-stationary objects like rain, leaves and snow. There are many applications of background modeling e.g. traffic monitoring, counting people, object tracking etc. In simple case, previous frame is considered to be the background frame for every frame. This way only the changes can be estimated, which is good for video compression, but not suitable for motion detection applications. Also this algorithm will not give good results for the objects moving very slowly.

Efficient algorithms are based on computing background from multiple frames. There are many algorithms proposed in the literature like Frame Difference method (Gonzalez and Wood 2002), Median Filter (Gonzalez and Wood 2002), Approximate Median (McFarlane et al 1995), Gaussian Mixture model (Stauffer and Grimson 1999), etc.

Background computation is followed by an update to accommodate challenging environment conditions. The video to be processed is loaded and background is computed using median filtering. In this technique, multiple frames are taken into consideration for calculating background. It was noted that for longer videos, if candidate frames are taken at intervals greater than one, the accuracy of both averaging and median filtering increases. This way decent background can be calculated.

$$B_{med}(x, y, t) = median(x, y, t-i)$$
$$i \in \{0, ..., n-1\} \quad (1)$$

Where $i \in \{0, ..., n-1\}$ are the frames of the video.

B. Foreground Segmentation

After computing the background frame, we subtract every frame of the video from this background frame. The background frame $B_{med}(x, y)$ calculated in (1) is subtracted from the original frames of the video to get the foreground frames named as $F_{MF}(x, y)$. It gives us the movement that is occurring in the video. Advantage of this approach is that any person that stops moving and was not in the background image is still picked up as an alien to the scene. But the drawback is that if a static person moves which was part of the background frame, its old position and the current position both are counted as pixels where movement has occurred. This artifact causes problem in the counting of the people in the video. Different artifacts caused by lighting condition changes, sensor noise are also picked up as small movements which are not desirable.

To get rid of the noise due to sensor, lighting conditions and shadows, we employed morphology. If we use the erosion on the output of subtraction then it eliminates the small artifacts or false positives due to sensor noise, giving out a cleaner image. A 3×3 erosion operator is applied to $F_{MF}(x, y)$ to get a new foreground frames called $F_{EM}(x, y)$. In this paper, we have proposed combination of techniques to get the best of techniques and improved the results so that better people counting can be achieved. For this purpose foreground segmentation is done with different techniques and texture analysis is done on combination of two techniques namely median filter and erosion operator after median filtering. Similarly $F_{edge}(x, y)$ is calculated based on edge detection through homogeneity and edge cancellation from the frames.

C. Feature Extraction

Once background frame is calculated and foreground is segmented from all frames of the video, different features are extracted from the foreground and their correlation with the ground truth is studied. Following are the description on the features,

Total Area occupied by Foreground (*FA*): All the frames after foreground segmentation are converted to binary image F_B having pixel values equal to 0 if it is part of background and 1 if it is part of foreground. Total Area occupied by the foreground is calculated as follows,

$$FA = \sum \sum F_F(x, y) \quad (2)$$

It is assumed that higher the value of ***FA***, higher will be the number of people in the frame. This feature set is calculated for three segmentation techniques namely, median filter of difference of frames, Edge detection with edge cancellation and foreground from background using erosion. Moreover, logical AND operation is performed on two images obtained from median filter of difference of frames and foreground from background using erosion methods to combine the good points of both techniques and provide common overlap of the two methods.

$$F_{overlap} = F_{MF}(x, y) \cap F_{EM}(x, y) \quad (3)$$

Where $F_{MF}(x, y)$ the foreground image using medians filter on frame difference and $F_{EM}(x, y)$ is the foreground image using erosion method. ***FA*** of the image $F_{overlap}$ is also calculated.

Contrast, Correlation, Energy, Homogeneity: In the next step, gray-level co-occurrence matrix (GLCM) is calculated from the frames $F_{overlap}$ converted into gray scale. GLCM is proposed by Haralick in 1979 (Haralick 1979) and now widely used in the texture analysis of the images. GLCM is the square matrix of size equals to the gray level of the frame. Its entries *p(i, j)* correspond to the number of times i^{th} gray level occurs near to the j^{th} gray level. Contrast, correlation, Energy and Homogeneity are calculated by the following formula,

$$Contrast = F_{con} = \sum_{i,j} |i-j|^2 \, p(i, j)$$

$$Correlation = F_{corr} = \sum_{i,j} \frac{(i-\mu i)(j-\mu j)\, p(i, j)}{\sigma_i \sigma_j}$$

$$Energy = F_{en} = \sum_{i,j} p(i, j)^2$$

$$Homogenity = F_{\text{hom}} = \sum_{i,j} \frac{p(i, j)}{1+|i-j|}$$

$$Entropy = F_{entropy} = -\sum p \times \log_2 p$$

Hence there are total nine features in the feature set F_{set} as described below,

$$\{F_{con}, F_{corr}, F_{en}, F_{\text{hom}}, FA_{MF}, F_{entropy}, FA_{EM}, FA_{edge}, FA_{overlap}\}$$

D. Classification

Neural networks (Fausett 1994) are widely used for pattern classification and function approximation. In this paper, we have used back-propagation neural network to perform the people counting based on the

features extracted from the texture analysis done in the previous sub-section. Figure 2 shows a framework to train the neural network. Back-propagation neural network is used with single layer of hidden neurons containing N hidden neurons. Weights of the neural networks are initialized randomly and features are fed to the input layer of the neural network and the output (number of people in the frame) is approximated by the neural network. This estimate of people count is compared with the true number of people in the frame and counting error is calculated which is used for the tuning of the weights of neural network.

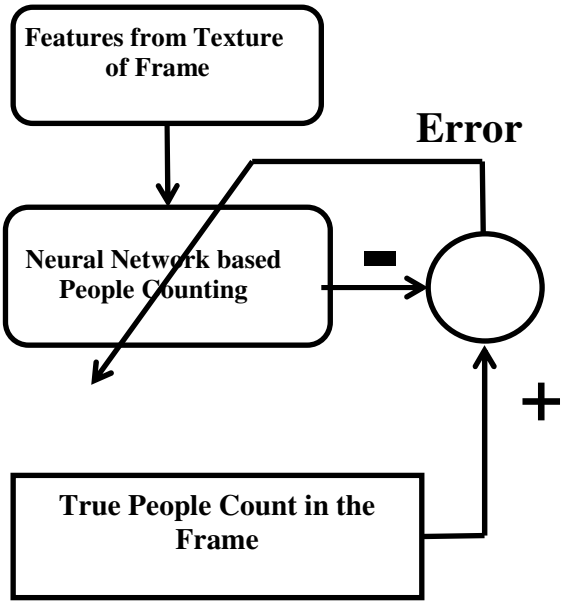


Figure 2: Training of Neural Network based people counting

For the tuning of the weights, Levenberg-Marquardt training algorithm is used. Once the neural network is trained by sufficient number of training patterns, it can be used to count the people in the unknown frames.

E. Video Data for Experimental Results

The experiments were performed on a video of pedestrians. The video had a resolution of 960 and 720 pixels along the horizontal and vertical axes, respectively. We have used 200 frames from this video for training and testing of the algorithm. Manual counting of pedestrians for each frame in both the training- and testing-datasets was performed to establish the ground truth. The manual counting was performed twice to correct the intra-rater variability (i.e., the variability which occurs even when the counting is performed by a single rater/individual). The video frames contain between 0 and 30 pedestrians.

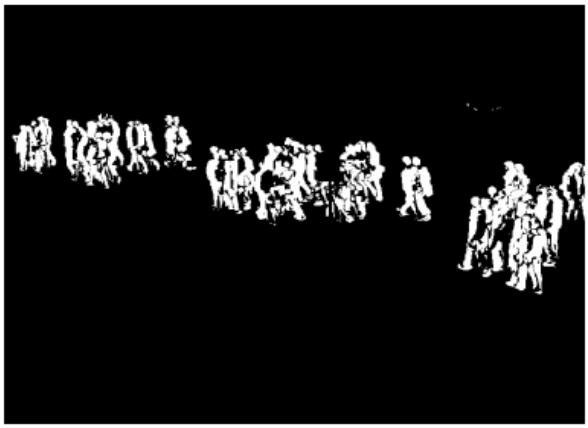
3. Experimental Results and Discussion

To validate our proposed framework, ground truth of the benchmark video is calculated and stored in the file for 200 frames. In the first step, background frame is calculated through median filter using a fixed number of frames buffered in the memory. This background frame is subtracted from the frames to segment the people moving in the video.

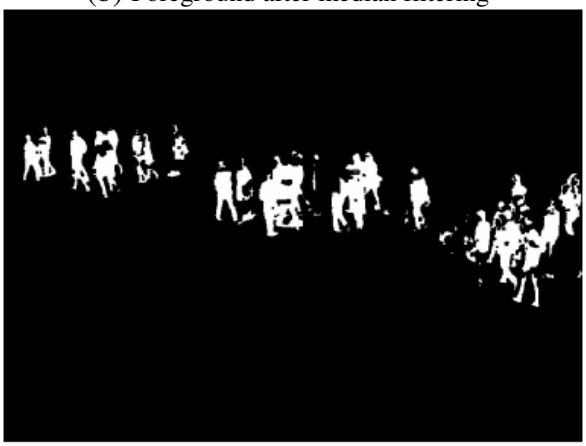
Figure 3: Some representation frames from the video**.**

Figure 4(a) shows a representative original frame of the video and figure 4(b) shows foreground frame obtained by subtracting the background frame. In this frame moving people are segmented but there are pixels representing the noise in the segmentation process. To remove the noise, erosion process is done to erode the noise from the foreground frame. Figure 4(c) shows foreground frame after erosion.

(a) Frame from Original video



(b) Foreground after median filtering



(c) Foreground using erosion

(d) Overlap image of (b) and (c)

(e) Foreground after Edge Detection using homogeneity with edge cancellation

Figure 4: Foreground extraction from the original video with different techniques

In the erosion process, the noise can be removed but some good pixels may also be removed.

We also applied logical AND operation and overlap frame is calculated using figure 4(b) and 4(c) and shown in figure 4(d). Texture analysis will be done on this overlap frame to extract the texture features like contrast, correlation, entropy etc. Another technique of foreground segmentation is used based on the edge detection using homogeneity and later cancelling these edges. Result is shown in figure 4(e). All of these frames will be used to extract the features in the next stage. As described in section 2(c), total of nine features are calculated from these frames shown in figure 4. To study the sensitivity of these features, every feature is plotted versus the ground truth (True number of people in the frame) in figure 5 (from the top left, feature 1 to feature 9 are plotted row wise).

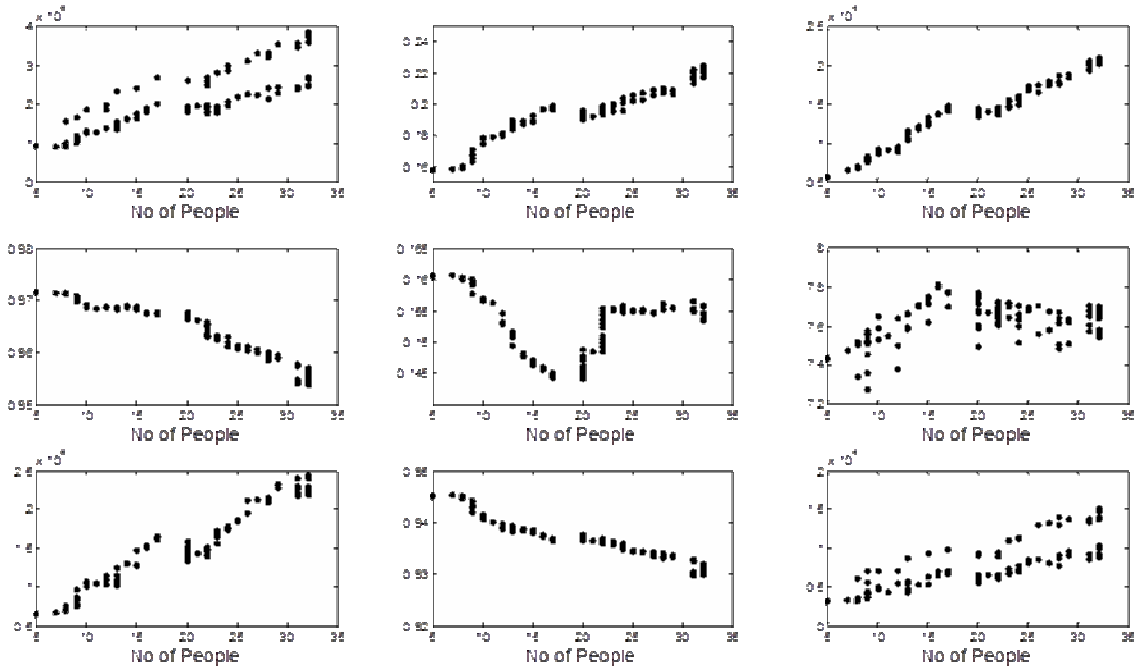


Figure 5: Correlation coefficient of the features with the ground truth (True number of people)

Correlation coefficient is calculated for all nine features with the true number of people in the frames and the values of correlation coefficient are also tabulated in table 1.

Table 1. Correlation coefficient of the features with true number of people in the frame

Feature Number	Correlation Coefficient
FA_{MF}	0.68
F_{con}	0.96
FA_{Edge}	0.99
F_{corr}	-0.93
F_{en}	-0.22
$F_{entropy}$	0.46
FA_{EM}	0.97
F_{hom}	-0.94
$FA_{overlap}$	0.68

It can be observed from the figure 5 and table 1 that some features are having very strong correlation with the number of people present in the frame whereas some features are not having good correlation with the number of people in the frame.

Based on this information, we have selected five features for the training of the neural network whose absolute value of correlation coefficient is more than 0.9. These features are stored in the new feature set F_{set}^{new}.

$$F_{set}^{new} = \{F_{con}, F_{corr}, F_{\text{hom}}, FA_{EM}, FA_{edge}\}$$

The new feature set F_{set}^{new} is calculated for all the frames of the video and divided randomly into training and testing set of 100 frames each. To find out the best number of hidden nodes, prediction error (in percentage) is calculated for different number of nodes on the training set as shown below,

$$Error_{\text{Pred}} = \frac{\left| PC_{pred} - PC_{True} \right|}{PC_{True}} \times 100$$

Where PC_{pred} is the people count predicted by the neural network and PC_{True} is the true people count in the frame.

Mean of the prediction error $Error_{\text{Pred}}$ is plotted in figure 6 for different number of hidden neurons starting from 1 to 100 at the step of 5.

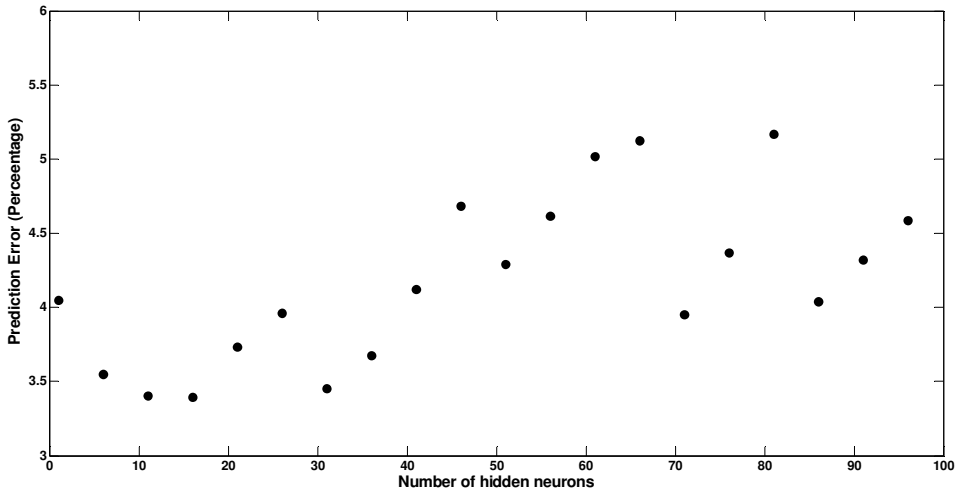


Figure 6: Prediction error in percentage for different number of hidden nodes in the neural network

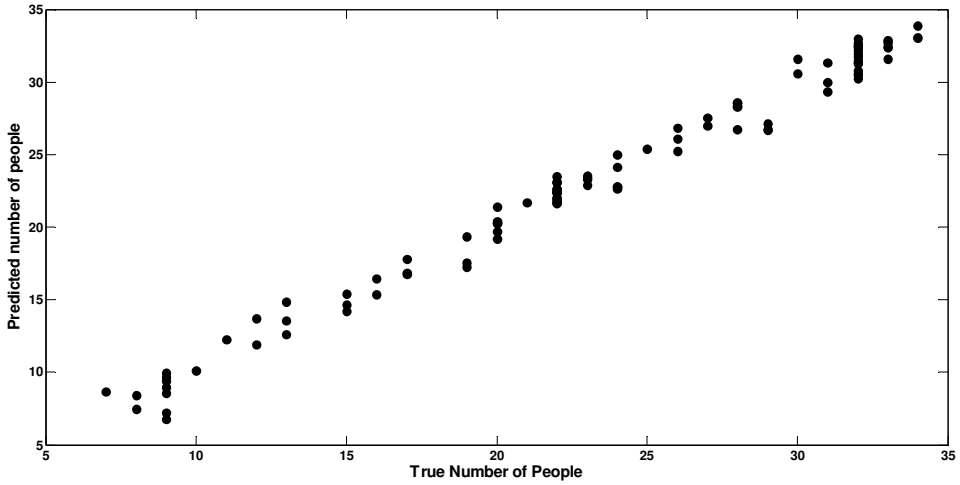


Figure 7: Plot of Predicted number of people and true number of people in the frames

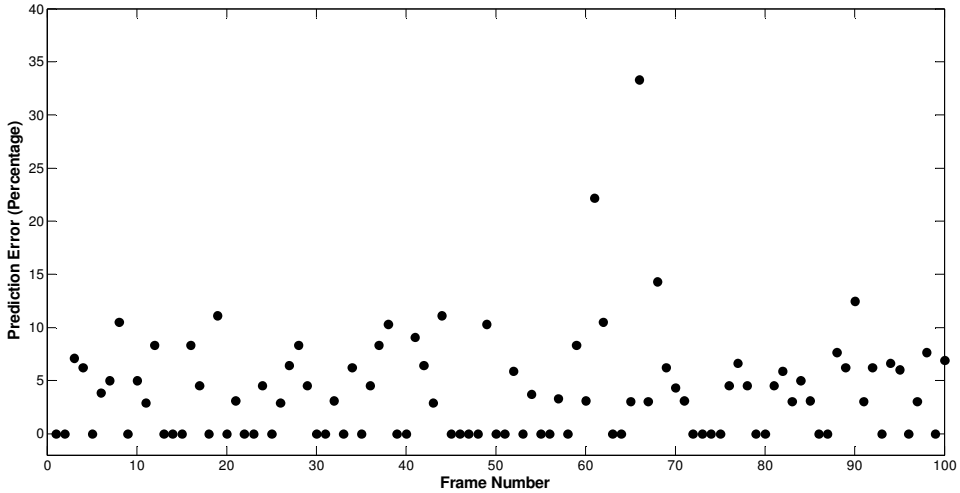


Figure 8: Prediction error in percentage for all testing frames

Thirty independent trials for each number of hidden nodes are done to remove the chance of trapping the learning algorithm of neural network in the local optimum value. From the figure 6, it is clear that optimal number of hidden nodes is 16 for which the prediction error is 3.39%. Hence number of hidden neurons of the neural network is set to be 16 for the testing data.

To test the performance of the neural network, prediction error of people counting in the testing data set is calculated. Whole features data is divided in to training and testing data (100 frames each) 20 times randomly to make the results statistically acceptable. Prediction error of the people counting after 20 runs is found to be 4.03%±4.10% on the testing data sets. For one testing data, predicted people count by neural network is plotted versus true number of people count in the frame in figure 7. It can be seen from the figure that both data are correlated strongly and a nice linear trend is observed in the figure.

Prediction error (in percentage) of people counting is also plotted in figure 8 and it can be seen from the figure that in most of the frames the error is below 5% counting error. These results shows efficacy of the proposed neural network based people counting from the video automatically. In the future work, we will study the effect of different lighting conditions will be studied for the proposed framework.

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 ".

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

References

1. Velastin SA, J. H. Yin JH, Davies AC, Vicencio-Silva MA, Allsop RE, Penn A. Analysis of crowd movements and densities in built-up environments using image processing. in Proc. IEE Colloquium Image Processing for Transport Applications, Aug. 1993.
2. Velastin SA, J. H. 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, London, U.K., 1994:127-132.
3. Zhao T, Nevatia R. Bayesian human segmentation in crowded situations. IEEE Conference on Computer Vision and Pattern Recognition, 2003;2:459-466.
4. Yoshinaga S, Shimada A, Taniguchi R. Real-time people counting using blob descriptor. Procedia Social and Behavioral Sciences, 2010;2:143-152.
5. Xiaohua L, Lansun S, Huanqin L. Estimation of crowd density based on wavelet and support vector machines. International Conference on Intelligent Computing, 2005.
6. Albiol A, Silla MJ, Albiol A, Mossi JM. Video analysis using corners motion analysis. In Proc. International Workshop on Performance Evaluation of Tracking and Surveillance (PETS2009), 2009:1-38.
7. Brostow G, Cipolla R. Unsupervised Bayesian detection of independent motion in crowds. In Proc. IEEE Computer Society Conference on Computer Vision & Pattern Recognition (CVPR2006), 2006;1:594-601.
8. 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.
9. 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.
10. Hashimoto K, Morinaka K, Yoshiike N, Kawaguchi C, Matsueda S. People count system using multi-sensing application. 1997 International conference on solid state sensors and actuators, 1997.
11. Davies AC, Yin JH, Velastin SA. Crowd monitoring using image processing. IEE Electronic and Communications Engineering Journal, 1995;7(1):37-47.
12. Roqueiro D, Petrushin VA. Counting people using video cameras. International Journal of Parallel, Emergent and Distributed Systems (IJPEDS), 2007;22(3):193-209.
13. Reisman P, Mano O, Avidan S, Shashua A. Crowd detection in video sequences, International Symposium on Intelligent Vehicles, 2004:66-71.
14. Marana A, Velastin S, Costa L, Lotufo R. Estimation of crowd density using image processing. Image Processing for Security Applications, IEE Colloquium, 1997;11:1-8.
15. Ma R, Li L, Huang W, Tian Q. On pixel count based crowd density estimation for visual surveillance. 2004 IEEE Conference on

Cybernetics and Intelligent Systems, 2004;1:170-173.
16. 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.
17. 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.
18. 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.
19. Huang D, Chow TWS, Chau WN. Neural network based system for counting people. IEEE Trans on Systems, Man, and Cybernetics, Part B, 2002;1(4):2197-2200.
20. Lo BPL, Velastin SA. Automatic congestion detection system for underground platforms. In Proc. Of the Int'l Symposium on Intelligent Multimedia, Video and Speech Processing, 2001:158-161.
21. Dan K, Doug G, Hai T. Counting Pedestrian in Crowds using viewpoint invariant training. In Proc. BMVC. 2005.
22. Stauffer C, Grimson WEL. Adaptive background mixture models for real-time tracking. In CVPR, 1999;2:2246.
23. Aguilera J, Thirde D, Kampel M, Borg M, Fernandez G, Ferryman J. Visual Surveillance for Airport Monitoring Applications. In Proc. of 11th Computer Vision Winter Workshop, 2006:5-10.
24. 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.
25. 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;28:299-308.
26. Gonzalez RC, Woods RE. Digital image processing, ISBN:9780201180756, Prentice Hall, 2002.
27. Stauffer C, Grimson WEL. Adaptive background mixture models for real-time tracking. In Proc. Computer Vision and Pattern Recognition, 1999;2:246-252.
28. McFarlane NJB, Schofield CP. Segmentation and tracking of piglets in images. Machine Vision and Applications, 1995;8(3):187-193.
29. Haralick RM. Statistical and structural approaches to texture, Proc. Of IEEE, 1979;67(5):786-804.
30. Fausett LV. Fundamentals of Neural Networks: Architectures, Algorithms, and Applications Prentice-Hall, 1994.

7/12/2012