Towards Real-Time Monitoring of the Hajj

Muhammad Baqui and Rainald Löhner
CFD Center/George Mason University
4400 University Avenue, MS 4c7, Fairfax, VA 22030, USA
mbaqui@gmu.edu; rlohner@gmu.edu

Abstract – An automated approach to explore the fundamental properties of high-density pedestrian traffic is outlined. The framework operates on video or time lapse images captured from surveillance cameras. For pedestrian velocity extraction, the framework incorporates cross-correlation based Particle Image Velocimetry (PIV) techniques. For pedestrian density estimation, the framework relies on the Machine Learning technique of the Boosted Regression Trees. The information collected from images in pixel coordinates are transformed to world coordinates with a pin-hole camera based projective transformation technique. The framework has been tested with high density crowd images acquired during the Muslim religious event, the Hajj. Accuracy and performance of the framework are reported.

***Keywords*:** Crowd Monitoring, Particle-Image Velocimetry, Machine Learning, Hajj

1. Introduction

Every year millions of Muslims congregate to perform the Hajj. Managing pedestrian safety and comfort for crowds of this size presents formidable challenges. As recently as 2015 hundreds of people lost their lives in an unfortunate accident during the Hajj [1]. Since surveillance cameras are widely used, studying video or time lapse photos from these cameras may provide valuable insights on the dynamics of high-density pedestrian traffic. The aim of this work is to enable automatic processing of these images and the extraction of quantifiable information. More specifically, the current work provides ways for obtaining velocity and density information from surveillance camera images. The framework relies on the Particle Image Velocimetry (PIV) [2] technique for pedestrian velocity extraction and a trained Machine Learning model for obtaining density. After obtaining velocity and density in the image/pixel coordinates, these are transformed to physical units (meters) in world coordinates through projective geometry (perspective correction).

The rest of the paper is organized as follows: in Section 2 a brief description is provided for the current state of research with image processing of crowd images. In Section 3 the theoretical aspects of the framework are discussed followed by Section 4, where the experiments and results are reported on the accuracy of the framework. In Section 5, some potential applications of the data are presented and finally conclusions are drawn on Section 6.

2. Background

Image processing and computer vision techniques have been used to analyze various aspects of pedestrian dynamics, namely walking behavior, crowd monitoring, head counting, trajectory extraction etc. In recent times, Maurin et al. [3] have constructed a crowd monitoring system based on optical flow, segmentation, and Kalman filter. Nedevschi et al. [4] also constructed a detection and collision avoidance scheme from video data based on Kalman Filtering. The results presented in both of these studies are promising. However, a major limitation of these frameworks is that they are not designed to handle high density crowds.

In order to obtain pedestrian density from a given image, the first step would be to get a headcount of the people in that image. Machine learning models have been shown to be effective in order to achieve

this. The crowd counting has often been formulated as a regression problem in machine learning. A regression model would output a predictive headcount once it is trained properly. Ma et al. [5] have developed a counting model based on Gaussian process regression. Idrees et al. [6] have constructed a Support Vector Regressor that has been trained with more than one feature (image gradients, Fourier peaks and interest point based samplings).

For obtaining pedestrian speed, Optical Flow [7] is arguably the most popular. The optical flow technique resolves the pedestrian velocities per pixel. As a result, the processing time becomes quite high for surveillance camera images which often have resolutions close to 5760x3840 pixels. The PIV technique that being widely used in the fluid dynamic community can be used to extract pedestrian velocities in a much faster rate. The PIV technique was developed in the mid 80's [8], and has recently found its use in other fields. For example, Vanlanduit et al. [9] have used PIV for metal fatigue experiments and Sveen et al. [10] applied it to the study of water waves. The application of PIV in audio speaker performance can be found in Rossi et al. [11]. From the review of relevant literature, it is apparent that by combining machine learning techniques with Optical Flow or PIV speed detection, one can conveniently create an automated framework that would monitor crowd properties and provide quantifiable information which could be used to make decisions on crowd management such as impending danger.

3. Materials and Methods

This section outlines a brief description of the theoretical aspects of the crowd monitoring framework. Crowd velocity extraction from PIV is discussed in section 3.1, crowd counting through boosted regression trees is discussed in section 3.2 and image-to-world coordinate transformation is discussed in section 3.3.

3.1 PIV

The PIV technique takes a sequence of time lapse photos that are being separated by a small time-gap (typically fractions of seconds). Initially the photos are divided into smaller blocks known as the interrogation spots. Afterwards, for each interrogation spot, cross correlation is performed. The cross-correlation computes component wise inner product. The inner product is generally computed in the frequency domain through convolution. Operating in frequency domain enables much faster processing time compared to regular (direct) correlation. In Figure 1, a sample input image and correlation surface can be seen.

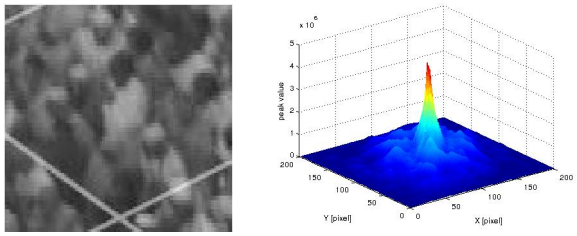


Fig 1. Input image and corresponding PIV correlation surface

3.2 Head counting

Head counting or pedestrian counting is the first step for the density estimation. The machine learning model formulated for head counting uses Histogram of Oriented Gradients (HOG) [12] as image features and manual annotations as ground truth. The machine learning model solves a regression problem following the gradient boosting algorithm [13]. Input to the machine learning model consists of image segments with pedestrians in it and output would be approximate counts of the number of people. To achieve the final counts, the machine learning model operates in two stages. In step 1, image features

are extracted via HOG and in the subsequent step, the regression model is trained with the HOG features and ground truth counts.

3.2.1 Feature Extraction via HOG

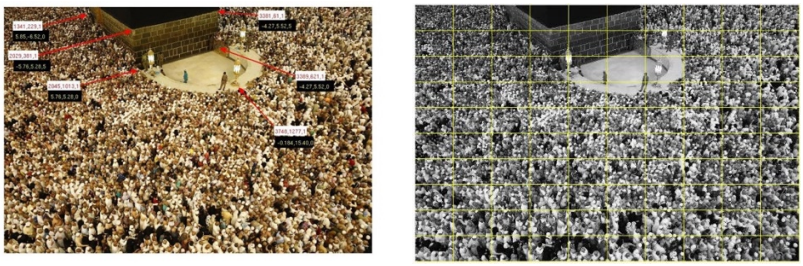
The HOG feature creates a histogram of images edges based on their orientation. The input image segment is first divided into 8x8 pixel blocks. For each block, image gradient is calculated. If the input image is an RGB image, it is first converted to Gray scale image (0-255 gray level values) in order to reduce the influence of illumination effects. Then gradients are calculated in a block wise fashion. Afterwards, these gradients are collected into 9 orientation bins. The final outcome is a histogram of this gradients. More information on its application to crowd counting can be found in [14].

3.2.2 Regression Model Construction

The regression model is constructed with HOG features of the input images and their corresponding ground truth counts. The model input has the form of ($[x_{11}, x_{12}, x_{13},.....,x_{1m}]$, y_1),($[x_{21}, x_{22}, x_{23},.....,x_{2m}]$,$y_2$).....($[x_{n1}$, $x_{n2}\ldots,x_{nm}]$, y_m), where n is the number of training samples (100 in this case) [Figure 2(b)] and m is the dimension of the HOG histogram (68600 in this case). The goal of the regression model is to formulate an approximate function F, ($\mathbf{x_{nm}}$− >y) that minimizes dissimilarity between the ground truth and the model prediction in a stage wise fashion by formulating trees (regression trees). After the model is being trained, if an input image is given, the model will first compute the HOG features of the input image and with the HOG features, the regression model approximates a head count of pedestrians in the image. More details of the approach can be found in [14] and [15].

3.3 Image to World Transformation

The pedestrian speed obtained through PIV comes in pixel coordinates. Also, for the density calculation, the counts obtained through Machine Learning need to be divided by the image area. As the images are not taken from an orthogonally posed camera, converting of these pixels to meters/centimeters becomes a challenge. The coordinate transformation technique involved here operated on a few landmark points for which pixel and world coordinates are known. In Figure 2 (a), these landmark points along with their pixel and world coordinates can be seen. Since the pedestrians are moving along a 2D plane, an equation of this plane is formulated in the image and world coordinates. Later, an intersection of any pixel point with this plane is determined. This intersection point is then converted to 3D coordinate with camera intrinsic parameters. These intersection points can be pixel locations of pedestrians or PIV displacements. More details of the approach can be found in [15].



(a) Landmark Points (b) Perspective cells

Fig 2. Landmark points and cells used for image to world transformation

4. Experiments and Results

In this section, the results are presented for the numerical experiments undertaken in order to investigate the performance of the framework in real world application. In section 4.1, the dataset used for the study is presented. In sections 4.2 to 4.4, accuracy results are presented for velocity extraction, head counting and image-to-world coordinate transformation.

4.1 Dataset

As the paper title implies, the focus of the study is the Muslim event, the Hajj. The Sahn area of the Hajj constitute the gathering (Figure 3) where people move circularly around the kaaba (black building) , this is called tawafs. The study focuses two different camera images. A sample frame from the second camera can be seen in Figure 3(a). These images are taken from Closed Circuit Television camera (CCTV) located at the facility. As part of the training and testing of the machine learning model, 600 image segments were manually annotated. Some annotations can be seen in Figure 3(b).

(a) Sample frame of dataset (b) Sample annotation

Fig 3. Sample frames of dataset and manual headcounting annotation

4.2 Velocity Extraction

The velocity vectors obtained from PIV processing of crowd images can be seen in Figure 4. Figure 4 (a) depicts the vectors of the entire image while Figures 4 (b) and (c) show vectors from two selected portions of Figure 4 (a). As can be seen from the magnified sections [Figure 4(b) and (c)] that the vectors are not streamlined. This is a result of high density. The predominant flow in this location is circular. However due to high density, some density waves appear that results in the chaotic patterns of the vectors.

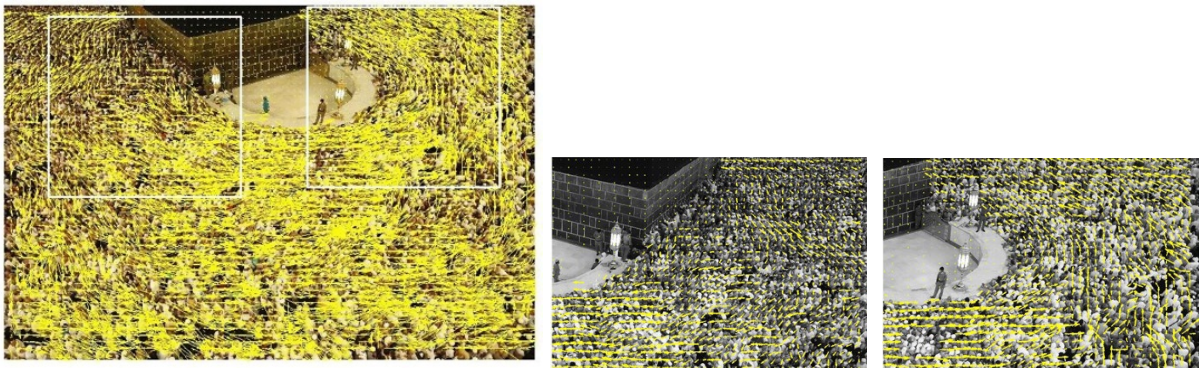


Fig 4. PIV velocity vectors of hajj crowds. Smaller images depict the vectors more clearly

To check the accuracy of the PIV velocity extraction, a number of methods have been undertaken. One of them is to manually track some random people at random locations of an image. In Table 1, the

results are listed for 6 random locations with PIV and manual tracking. It was found that the maximum error was about 16%.

4.3 Head counting Results

The head counting from images are needed to obtain density. In Table 2, the results are presented for the Machine Learning model vs the ground truth counts for 6 different images of both datasets. As can be seen from the table (Table 2) the countings are close to each other.

Table 1 PIV displacement and ground truth comparison

Actual dx (pixel)	PIV dx (pixel)	Actual dy (pixel)	PIV dy (pixel)	%Error
-3	-2.52	3	2.73	12.43
-3	-3.89	5	4.08	3.32
-8	-6.87	3	2.39	14.86
-5	-4.28	4	4.14	7.08
4	3.08	-2	-2.14	16.13
8	7.55	2	1.72	6.09

Table 2. Comparison between Machine Learning count and ground truth count

Test set number	Frame number	Machine Learning model count	Ground truth count	%Error
1	5	2949	2910	1.34
1	6	3024	3263	7.32
1	8	2981	2816	5.58
2	4	4449	4451	0.0004
2	10	4564	4686	2.60
2	20	4423	4729	0.6

As mentioned earlier in section 3.2, for head counting, the input image is being divided into 100 smaller sub images i.e. image cells [Figure 2(b)]. The Machine Learning model provides predictive counts for each of this smaller image cells. In Figure 5, the blue dots represent the predictive headcount from Machine Learning model while the error bars indicate difference between the ground truth counts and the predictive counts. It can be seen that from cells 0 to 50, the headcounts vary quite a bit. This is because of the camera angle (camera perspective). The effect diminishes in the cells from 50-99. However, Table 2 indicates that the cumulative counts are close to the ground truth counts. So, the perspective is not severely affecting the overall counts.

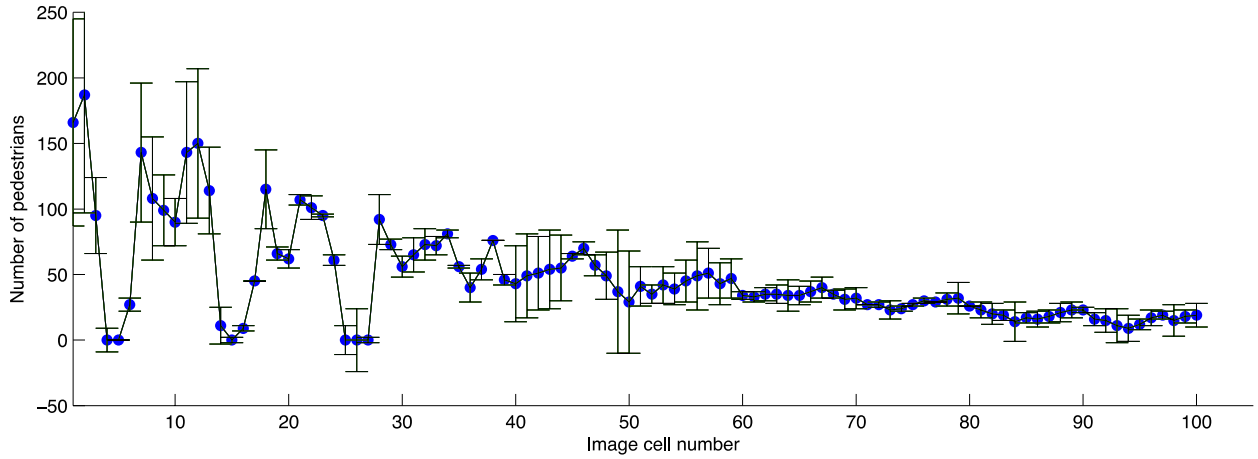


Fig 5. Machine Learning model and ground truth count difference for one test image

4.4 Coordinate transformation results

The results of this transformation process for 4 different coordinate points can be seen in Table 3. It can be seen that the projected points are not very far from the world coordinates.

Table 3. Comparison between transformed image coordinates (to world) and actual world coordinates

World x (m)	World y (m)	World z (m)	Projected x (m)	Projected y (m)	Projected z (m)	Difference in person width
1.77	15.35	0.0	2.54	14.65	0.0008	2.08
-1.84	15.40	0.0	-0.183	15.39	0.0008	2.02
5.85	-6.52	0.0	4.09	-6.31	0.0008	3.54
-4.27	5.52	0.0	-4.87	5.18	0.0008	1.37

4.5 Processing Time:

Compute time plays a key role in real-time processing applications. The framework outlined in this study has two components (PIV processing and Head counting) where timings are critical. To process two frames of the second dataset (Figure 3) took 32.79 seconds. The timing includes the PIV processing and head counting combined. The experiments are performed in a 4-core laptop with 1.8 GHz processor and 8 GB main memory. So, in terms of compute power requirement, the framework is not very demanding. However, as more cameras are incorporated in the processing, the demand for compute power will go up. GPU processing may offer a simple solution to this.

5. Application

When fundamental properties (density and velocity) of a crowd are known they can be incorporated into a fundamental diagram. Furthermore, they can also be used to obtain pedestrian distribution in future time. These applications are briefly explained in this section.

5.1. Fundamental Diagram

The fundamental diagram (speed vs density plot) for images from dataset 2 are shown in Figure 6. The fundamental diagrams are rare to find for high density flows considering the risks associated in conducting experiments in such conditions.

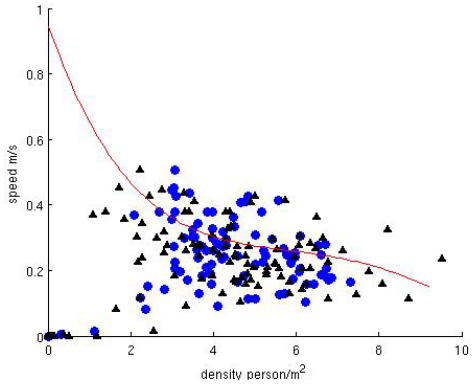


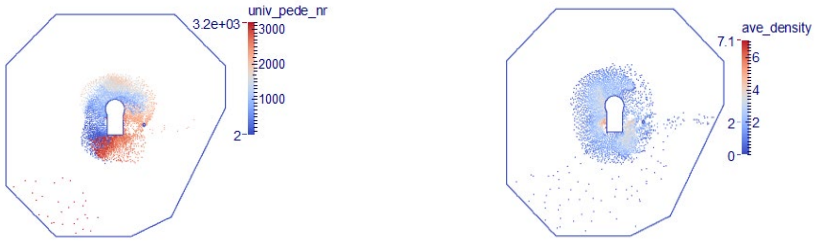
Fig 6. The fundamental diagram obtained from dataset 2 images. Blue dots are ground truth density and black triangles are predictive density. The solid line is from Predechenskii and Milinski [16]

The proposed framework can provide as an alternative approach in obtaining empirical results for high density crowds. The fundamental diagram of Figure 6 also graphically outlines the difference of density values obtained from the Machine Learning model and ground truth counts.

5.2. Predicting Future State of Crowd

Microscopic models such as PEDFLOW [17] take pedestrian density and speed as input and can approximate pedestrian distribution at future times. The current framework can enable the microscopic models to approximate a more accurate distribution of pedestrians through a more accurate input. In Figure 7, three images of the Kaaba premises are shown that gives a whole 360-degree coverage of the facility. The PEDFLOW approximation of pedestrian distribution and pedestrian density at 4 seconds in future can be seen in Figure 8. Although the approach is currently in its infancy, such an application possesses a great promise in high density crowd monitoring and accident prevention.

Fig 7. Input images processed with PIV for speed and machine learning for density as input to PEDFLOW



(a) Pedestrian distribution (b) Pedestrian density

Fig 8. The PEDFLOW simulation results at 4 secs in future

6. Conclusions and Outlook

The study addresses two critical challenges of high-density pedestrian traffic: real-time monitoring and estimation of future state (accident/congestion). In the case of real-time monitoring, the proposed PIV technique can be considered as a faster alternative of optical flow; while for future state estimation, the PEDFLOW model combined with inputs from PIV (speed) and machine learning model (density) provides a useful tool to the safety personnel managing/monitoring the crowd. The accuracy of the framework is reported for the crowd of Makkah. The main challenge lies in the scarcity of ground truth datasets in order to make the model more general. As of right now, there are only a handful of datasets for machine learning based crowd counting and their generalization ability is limited. An alternative to the scarcity of datasets could be to model individual crowd events separately and study them individually. This the route that the authors have taken in this study.

References

[1] S. Almukhtar and D. Watkins, "How One of the Deadliest Hajj Accidents Unfolded," *The New York Times*, 05-Sep-2016.
[2] R. J. Adrian and J. Westerweel, *Particle Image Velocimetry*. Cambridge University Press 558, 2010.
[3] B. Maurin, O. Masoud, and N. P. Papanikolopoulos, "Tracking all traffic: computer vision algorithms for monitoring vehicles, individuals, and crowds," *IEEE Robot. Autom. Mag.*, vol. 12, no. 1, pp. 29–36, Mar. 2005.
[4] S. Nedevschi, S. Bota, and C. Tomiuc, "Stereo-Based Pedestrian Detection for Collision-Avoidance Applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 3, pp. 380–391, Sep. 2009.
[5] Z. Ma and A. B. Chan, "Crossing the Line: Crowd Counting by Integer Programming with Local Features," *IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 1063–69/13, pp. 2535–2546, 2013.
[6] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, "Multi-source multi-scale counting in extremely dense crowd images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2547–2554.
[7] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proc. Int. Jt. Conf. Artif. Intell.*, pp. 674–679, 1981.
[8] R. J. Adrian and C. S. Yao, "Development of Pulsed Laser Velocimetry (PLV) For Measurement of Turbulent Flow," in *In Symposium on Turbulence. X.B. Reed Jr., G.K. Patterson ed*, 1984, pp. 170–184.
[9] S. Vanlanduit, J. Vanherzeele, R. Longo, and P. Guillaume, "A digital image correlation method for fatigue test experiments," *Opt. Lasers Eng.*, vol. 47, no. 3, pp. 371–378, 2009.
[10] J. K. Sveen and A. E. Cowen, "Quantitative Imaging Techniques and Their Application to Wavy Flows, In PIV and Water Waves," *World Sci.*, 2004.
[11] M. Rossi, E. Esposito, and E. P. Tomasini, "PIV Application to Fluid Dynamics of Bass Reflex Ports," in *Particle Image Velocimetry*, Springer, 2007, pp. 259–270.
[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005, vol. 1, pp. 886–893.
[13] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Ann. Stat.*, pp. 1189–1232, 2001.
[14] M. Baqui, "Automated Monitoring of High Density Crowd Events," 2018.
[15] P. Dollár, *Piotr's image and video Matlab Toolbox (PMT),https://pdollar.github.io/toolbox/*. 2013.
[16] V. M. Predtechenskii and A. I. Milinskii, *Planning for foot traffic flow in buildings*. National Bureau of Standards, US Department of Commerce, and the National Science Foundation, Washington, DC, 1978.

[17] R. Löhner, "On the Modeling of Pedestrian Motion," *Appl Math Model.*, vol. 34, no. 2, pp. 366–382, 2010.